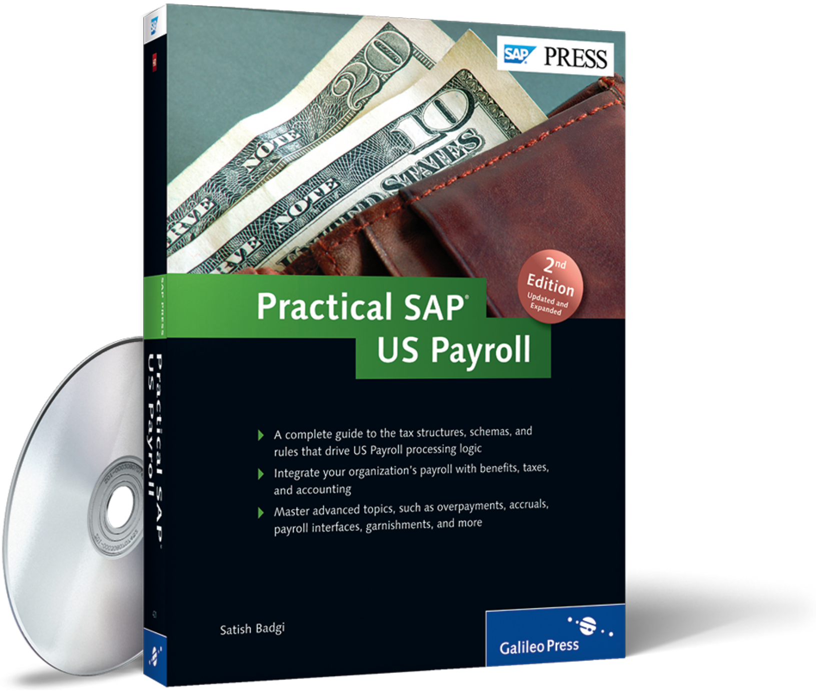
Satish Badgi

Practical SAP® US Payroll


SAP PRESS
2nd Edition
Updated and Expanded
Practical SAP®
US Payroll
A complete guide to the tax structures, schemas, and rules that drive US Payroll processing logic
Integrate your organization's payroll with benefits, taxes, and accounting
Master advanced topics, such as overpayments, accruals, payroll interfaces, garnishments, and more
Satish Badgi
Galileo Press
Practical SAP® US Payroll

Galileo Press®

Bonn • Boston

Contents at a Glance

1 Objectives and Purpose 23
2 US Payroll Infotypes: Master Data and Transaction Data 33
3 Wage Types 61
4 Schemas and Rules 125
5 Benefits Integration 175
6 Time Management and Payroll Integration 201
7 Garnishments 223
8 Tax Processing 241
9 Advanced Topics 279
10 Cutover, Implementation, and Post-Go-Live 325
A US Payroll Schema U000 363
B Wage Type Template 375
C Commonly Used Technical Wage Types 379
D Commonly Used Model Wage Types 409
E Commonly Used Processing Classes and Specifications 427
F Useful Payroll Operations 439
G The Author 449

Contents

Preface 15

1 Objectives and Purpose 23

1.1 What Makes US Payroll Unique? 24
1.1.1 Functional Uniqueness 24
1.1.2 Technical Uniqueness 27
1.2 The U.S. Payroll Process 29
1.3 Components of SAP US Payroll 30
1.4 Summary 32

2 US Payroll Infotypes: Master Data and Transaction Data 33

2.1 Master Data: Conceptual Clarity 33
2.2 Retroactive Accounting 34
2.3 Employee Master Data for US Payroll 36
2.3.1 Infotype 0007: Work Schedules 40
2.3.2 Infotype 0008: Basic Pay 40
2.3.3 Infotype 0009: Bank Details 41
2.3.4 Infotype 2010: Employee Remuneration 42
2.3.5 Infotype 0207: Residence Tax 43
2.3.6 Infotype 0208: Work Tax 43
2.3.7 Infotype 0167: Health Plans 44
2.3.8 Infotype 0057: Membership Fees 44
2.4 Employee Transaction Data: US Payroll 45
2.4.1 Infotype 0014: Earnings/Deductions Transaction Data ... 49
2.4.2 Infotype 0169: Savings Plans 50
2.4.3 Infotype 0015: Additional Payments and Deductions 50
2.4.4 Infotype 0267: Bonus and Off-Cycle 51
2.4.5 Infotype 0011: External Bank Transfer 52
2.4.6 Infotype 0210: Changes to W-4 Data 53
2.4.7 Infotype 0234: Additional Withholding Info 53
2.4.8 Infotype 0235: Other Taxes 54
2.4.9 Infotype 0554: Hourly Rate Per Assignment 55
2.4.10 Infotype 0027: Cost Center Distribution 55

2.5 Pre-Payroll Data Verification 56
2.6 Summary 59

3 Wage Types 61

3.1 Overview of a U.S. Paystub and the Concept of Wage Types 61
3.2 Lifecycle of a Wage Type and U.S. Wage Type Catalog 64
3.2.1 Wage Type Categories 68
3.2.2 Three Elements of a Wage Type 69
3.2.3 Wage Type Catalog 71
3.2.4 Wage Type Tables 74
3.3 Step-by-Step Configuration of a Wage Type 74
3.3.1 Earnings 75
3.3.2 Deductions 83
3.3.3 Taxes 89
3.3.4 Net Pay 93
3.3.5 Wage Type Configuration Tool 95
3.4 Testing Wage Types 97
3.5 Advanced Topics on Wage Types 101
3.5.1 Handling Cumulations 101
3.5.2 Setting Up Payment Models 103
3.5.3 Priority and Arrears 104
3.5.4 Posting of Wage Types 109
3.5.5 Valuation of Wage Types 122
3.6 Summary 123

4 Schemas and Rules 125

4.1 Examining the U.S. Payroll Driver (RPCALCU0) 125
4.2 Working with U.S. Schema U000 127
4.2.1 Schemas 130
4.2.2 Subschemas 130
4.2.3 U.S. Subschemas 132
4.3 Why, When, and How to Change the U.S. Schema 143
4.3.1 Copy and Modify Schema U000 using the Schema Editor 145
4.3.2 Running an Error-Free Schema 147
4.4 Overview of the U.S. Public Sector Schema Functionality 152
4.4.1 NRA: Non-Resident Alien Processing 153

4.4.2 Savings Bonds Processing 154
4.4.3 Public Sector Savings Plans: 403(B), 457(B) 154
4.4.4 Employer Benefits and Tax Allocation Rules 155
4.5 Overview of Non-Profit Organization (NPO) Payroll 155
4.6 Writing Rules in US Payroll 156
4.6.1 The Rules Editor 157
4.6.2 Creating Simple Rules 159
4.6.3 Creating Complex Rules 162
4.6.4 Running Error-Free Rules 170
4.7 Summary 173

5 Benefits Integration 175

5.1 Integration of US Benefits with US Payroll 175
5.1.1 Input 176
5.1.2 Processing 177
5.1.3 Output 177
5.2 Health Plan Integration 179
5.3 Savings Plan Integration 183
5.4 Insurance Plan Integration 190
5.5 Flexible Spending Account Integration 195
5.6 Sending Data and Remittance to Benefits Providers 197
5.6.1 Health Plans 197
5.6.2 Insurance Plans 198
5.6.3 Savings Plans 198
5.6.4 Sending Remittance 198
5.6.5 Sending Data 199
5.7 Summary 199

6 Time Management and Payroll Integration 201

6.1 Definitions 201
6.2 Time-Payroll Integration Infotypes 204
6.3 Time and Payroll Integration Scenarios 205
6.3.1 Scenario 1: SAP CATS + SAP Time Evaluation + SAP Payroll 207
6.3.2 Scenario 2: External Time Clocking + External Time Evaluation + SAP HCM and SAP Payroll 210

6.3.3 Scenario 3: External Time Clocking + SAP HCM and SAP Time Evaluation + SAP Payroll ... 212
6.3.4 Scenario 4: External Time Clocking + SAP CATS and SAP Time Evaluation + SAP Payroll ... 214
6.4 Challenges of Time Integration with Payroll ... 214
6.4.1 Using Custom Infotypes ... 214
6.4.2 Override Jobs ... 215
6.4.3 Override Cost Centers ... 215
6.4.4 Reconciliation Between External Time Clocking and SAP Payroll ... 216
6.5 Time Schemas ... 217
6.6 FLSA and Rate Calculation Impact ... 219
6.6.1 Minimum Wage ... 220
6.6.2 Overtime ... 220
6.6.3 Tracking of Hours Worked and Record-Keeping ... 220
6.7 Summary ... 221

7 Garnishments ... 223

7.1 Configuring Garnishments ... 224
7.2 Disposable Net Income ... 228
7.2.1 Definition of Disposable Net Income ... 228
7.2.2 Example ... 230
7.3 Managing Wage Types for Garnishments ... 232
7.3.1 Garnishment Deduction Wage Types ... 232
7.3.2 Impact of Other Wage Types ... 233
7.4 Garnishment Subschema ... 233
7.4.1 Rule UGIT ... 234
7.4.2 Rule UGDN ... 236
7.4.3 Function UGARN ... 238
7.5 Summary ... 239

8 Tax Processing ... 241

8.1 Introduction to U.S. Tax Processing ... 241
8.1.1 Tax Infotypes ... 242
8.1.2 Earnings and Deductions ... 244
8.1.3 Relevant Processing Classes for Wage Types ... 245

8.2 Tax Models 247
8.2.1 Tax Authorities 247
8.2.2 Tax Areas 248
8.2.3 Tax Types 249
8.2.4 Tax Classes 250
8.2.5 Tax Models 250
8.3 Year-End Tax Adjustments and Workbench 255
8.3.1 Starting the Year-End Workbench 257
8.3.2 Running Payroll 262
8.3.3 Verification 264
8.4 Overview of the Tax Reporter 265
8.4.1 Preparation Work 265
8.4.2 Quarterly Reports 266
8.4.3 Annual Reports 268
8.5 Tax Services 270
8.6 BSI Tax Factory 271
8.6.1 Installing BSI Tax Factory 9.0 272
8.6.2 Configuration of SAP for BSI 273
8.6.3 How to Apply BSI TUBS 274
8.6.4 Tax Locator 274
8.6.5 Handling Reciprocity 275
8.7 Summary 277

9 Advanced Topics 279

9.1 Overpayments and Claims Handling 279
9.1.1 Why Claims Are Generated 280
9.1.2 Identifying Claims in Payroll 282
9.1.3 Strategy with Claims 284
9.1.4 Clearing Claims: Payroll-Forgiven 284
9.1.5 Clearing Claims: Payroll-Not-Forgiven 285
9.1.6 Clearing Claims: Accounts Taxes (Employee Pays Single Check) 286
9.1.7 Clearing Claims: Accounts Taxes (Employee Pays by Payment Plan) 288
9.2 New Overpayments Functionality 290
9.2.1 Differences Between Old and New Functionality 290
9.2.2 Running New Overpayments Functionality 292

9.3 Accruals 298
9.3.1 Accruals Configuration 299
9.3.2 Accruals Process and Posting 302
9.4 Interfacing with Third-Party Payroll Systems: Gross to Net 306
9.5 Third-Party Remittance (Accounts Payable Processing) 312
9.5.1 Required Configuration 313
9.5.2 Financial Account Management 314
9.5.3 Running a Third-Party Process 316
9.6 Expatriate Payroll 319
9.6.1 Scenario 1: Payroll in SAP US Payroll for an Expatriate ... 319
9.6.2 Scenario 2: Payroll in SAP US Payroll and SAP Germany Payroll for an Expatriate 320
9.7 Implementation Tips for US Payroll 321
9.7.1 Go-Live Calendar 321
9.7.2 Volumes and Complexity 323
9.7.3 Resource Planning and Knowledge Transfers 323
9.8 Summary 324

10 Cutover, Implementation, and Post-Go-Live 325

10.1 Implementation Lifecycle 325
10.1.1 Structure to Discuss Payroll Implementation and Cutover 326
10.1.2 Implementation Challenges for Payroll 327
10.2 Cutover Planning 331
10.2.1 Configuration Teams 332
10.2.2 Payroll Operations 333
10.2.3 Interface Management 333
10.2.4 Training and Change Management 334
10.2.5 Finance and Other Integrated Applications 335
10.2.6 Security and Roles 335
10.2.7 Legacy Teams 335
10.2.8 Technical Teams 336
10.3 Typical Cutover Plan 336
10.3.1 Step 1: Pre-Cutover Activities and Tasks 337
10.3.2 Step 2: Cutover of Configuration 337
10.3.3 Step 3: Conversions 338
10.3.4 Step 4: Business Cutover 339

10.3.5 Step 5: Go-Live 340
10.3.6 Step 6: Post-Go-Live 341
10.3.7 Typical Implementation Challenges 341
10.4 Testing 342
10.4.1 System Integration Test 343
10.4.2 Parallel Testing 345
10.5 Post-Go-Live Challenges for Payroll 352
10.5.1 Housekeeping Activities 355
10.5.2 Best Practices to Maintain Wage Types, Schemas, and Rules 358
10.6 Summary 359

Appendices 361

A US Payroll Schema U000 363
B Wage Type Template 375
B.1 Part 1: To Be Completed by Requestor 375
B.2 Part 2: To Be Completed by Person Configuring 376
B.3 Approvals and Testing 378
C Commonly Used Technical Wage Types 379
D Commonly Used Model Wage Types 409
E Commonly Used Processing Classes and Specifications 427
F Useful Payroll Operations 439
G The Author 449

Index 451

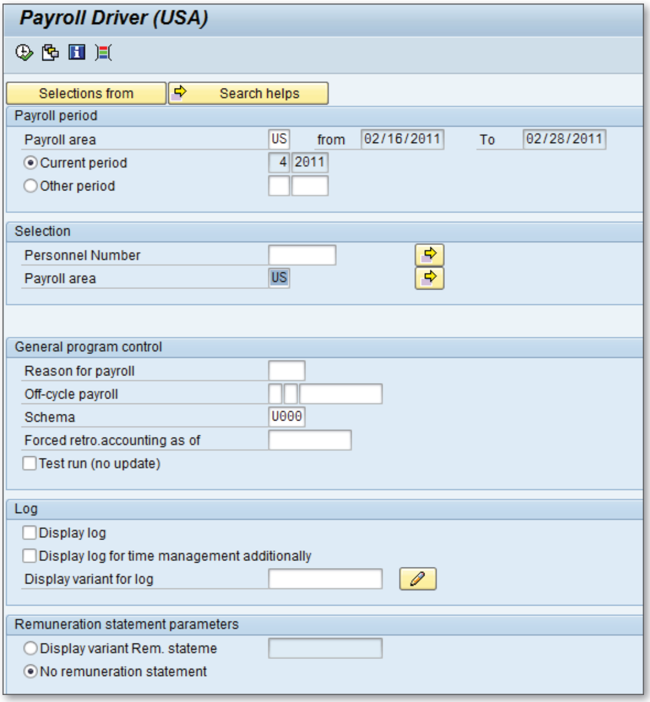
US Payroll schema U000, with its associated subschemas and rules, forms the core of the US Payroll processing system. Though writing rules is one of the challenging areas in SAP Payroll, it is a much-required skill to explore the flexibility of US Payroll. This chapter explains the functionality of the U.S. schema and presents examples to help you write your own rules.

4 Schemas and Rules

This chapter forms the heart and soul of the book. Though many users agree that there is a certain "fear factor" associated with writing schemas and rules in US Payroll, this chapter will bravely discuss US Payroll schema U000, which will be followed by the subschema discussion. The chapter discusses the modification of schemas and rules, including the runtime environment. It uses many practical examples to support rule-writing guidelines and provides guidance for future rule-writing initiatives. Dealing with schemas and rules requires a lot of practice, and this chapter will help you overcome those initial fears so you can immediately create your own custom rules using the examples. The US Payroll schema consists of a series of statements that control the logic for payroll calculations and the payroll process. Rules are an integral part of the schema and normally contain arithmetical and logical operations (such as multiplication and division). You can use these operations to manipulate the rate, number, and amounts in wage types. Let's dive in and examine the U.S. payroll driver.

4.1 Examining the U.S. Payroll Driver (RPCALCU0)

Before beginning the schema discussion, it's important to understand where and how the schema is used in the payroll process. The payroll driver is used whenever you run the payroll, either in simulation mode or start mode (as per the status of the payroll control record). *Payroll driver* is just another term for an ABAP program, RPCALCU0 (the *U* in the name of this program refers to the U.S. payroll driver). Figure 4.1 shows the opening screen of the driver as you'll see it in your own system.



Payroll Driver (USA)

Payroll period

Payroll area US from 02/16/2011 To 02/28/2011

● Current period 4 2011

○ Other period

Selection

Personnel Number

Payroll area US

General program control

Reason for payroll

Off-cycle payroll

Schema U000

Forced retro.accounting as of

□ Test run (no update)

Log

□ Display log

□ Display log for time management additionally

Display variant for log

Remuneration statement parameters

○ Display variant Rem. stateme

● No remuneration statement

Figure 4.1 RPCALCU0: US Payroll Driver

The payroll driver helps you with several tasks, including:

- Managing the payroll process based on the status of the payroll control record (start/correction/exit)
- Managing off-cycle and bonus runs, in addition to regular payroll runs
- Performing forced retroactive payroll runs
- Using a custom schema for payroll calculations based on your own requirements

Since the discussion here is about schemas, let's focus on using custom schemas. Custom schemas give you tremendous flexibility for payroll calculations. You can also use different schemas to run different payroll areas. For example, if you have a weekly payroll for hourly employees and a biweekly payroll for salaried employees, then you will have two payroll areas: one for weekly and one for biweekly. You can create two schemas—one for each area. (The payroll areas should be kept simple and straightforward for overall ease of maintenance.) For example, if different geographic locations have different payroll rules and scenarios, you'll need to create separate payroll areas for each of them. Of course, you can also use one schema for both payrolls and create separate rules for salaried and hourly employees. If the two payrolls are quite different from each other in terms of policies and rules, then you may need separate schemas. For example, a company acquires another company and wants to create a separate schema because of large differences in rules and calculations. You will learn how to create rules later in Section 4.5. The example in Figure 4.1 relates to the payroll driver, which uses schema /Z00 to run the payroll. That means the logic and rules coded in schema /Z00 are used by the payroll run.

During testing, you have the ability to turn logging on so you can dissect the schema and learn more about any errors and issues that occur. Examples will be discussed later in this chapter. For now, don't set this flag to on for a full payroll run because it can create too large a log and may create a program dump if the output becomes too large. This would happen if, for example, your payroll contained thousands of employees.

At this stage, it is important for you to read through the standard SAP documentation on payroll control record topics if you are not familiar with the different possible statuses of the payroll control record. It is essential for you to have this knowledge before proceeding to the next section, which dives into the U.S. schema (U000) and walks you through the important subschemas.

Tip

The SAP documentation is available online at *http://help.sap.com/*.

4.2 Working with U.S. Schema U000

Though it is possible that some readers have already heard of database schemas or XML schemas, the most basic question to answer is *What is a schema?* The US Payroll schema consists of a set of statements that drive the logic of payroll processing. As the word suggests, it is a defined plan to run the payroll.

The payroll schema helps the payroll driver with calculation rules, functions, and the overall flow of processing of employees in a payroll. The schema gives you the ability to “arrange” the processing, as you will see in the examples later in this chapter. Imagine that in your legacy payroll system, for example, a single payroll program is used to control the processing, which caused you to write complicated logic to manipulate calculations, manage union rules and deductions, and so on. Now, thanks to schemas, rules, and wage types in SAP Payroll, you don't have to write and maintain complex legacy code. While the payroll driver serves as the engine to run your payroll, the schema gives you the flexibility of "configuring" the rules. Figure 4.2 shows a concept diagram to explain the various parts of a schema. Consider it a bill of materials for a schema with all its subassemblies and parts. Notice that schemas can contain subschemas, and subschemas in turn can contain more subschemas. However, subschemas can be generally referred to as schemas as well. Figure 4.2 explains the relationships between schemas and subschemas, functions, rules, and operations. Don't worry about the actual examples used in Figure 4.2; for now, what's important is that you understand the basic concepts and relationships.

Note

SAP has provided a specific schema for the US Public Sector and also one for non-profit organizations. You will find the non-profit payroll in the SAP Payroll application menu. We will limit our discussion to the generic U.S. schema, but differentiators of the US Public Sector schema are listed in Section 4.1.

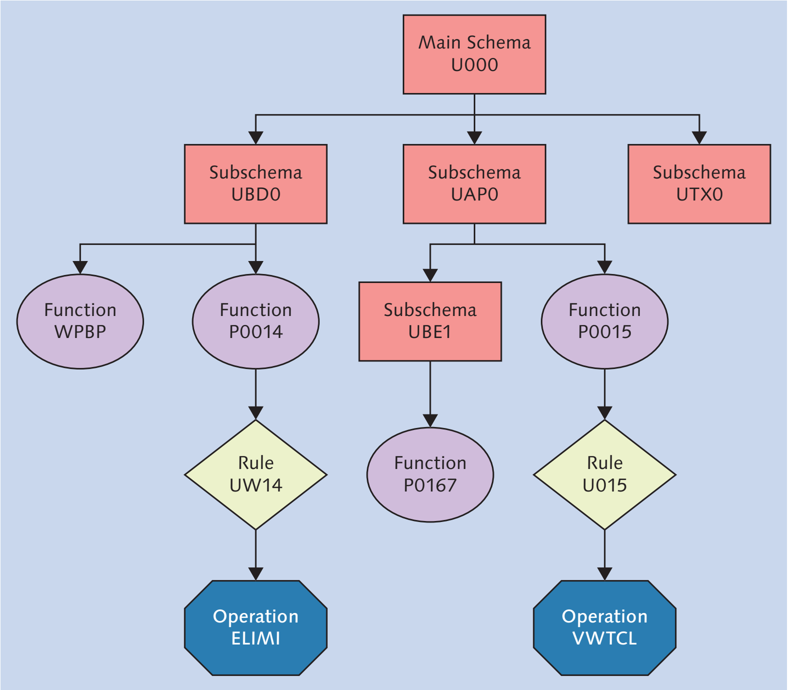
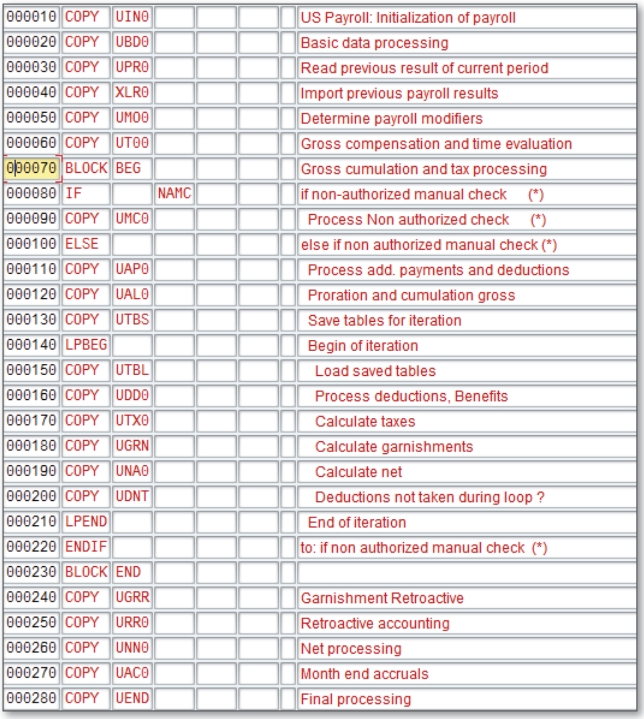


Figure 4.2 Schema and Subschema Tree Structure

Figure 4.2 presents a tree structure to show how the main schema, subschemas, functions, rules, and operations relate to each other. Treat it as an *ecosystem* of the payroll schema, and you will notice that the ecosystem depends on its elements to function properly. The following relationships are shown in Figure 4.2:

- The main schema consists of many subschemas.
- Both schemas and subschemas consist of many functions.
- Functions consist of many rules.
- Rules consist of many operations.

Together, these elements form the aforementioned schema ecosystem. Now, use Transaction PE01, and open U.S. schema U000, as shown in Figure 4.3. You will learn more about the Schema Editor and Transaction PE01 later in this chapter. For now, it's important for you to become familiar with the look and feel of the schema.



000010	COPY	UIN0					US Payroll: Initialization of payroll
000020	COPY	UBD0					Basic data processing
000030	COPY	UPR0					Read previous result of current period
000040	COPY	XLR0					Import previous payroll results
000050	COPY	UM00					Determine payroll modifiers
000060	COPY	UT00					Gross compensation and time evaluation
000070	BLOCK	BEG					Gross cumulation and tax processing
000080	IF		NAMC				if non-authorized manual check (*)
000090	COPY	UMC0					Process Non authorized check (*)
000100	ELSE						else if non authorized manual check (*)
000110	COPY	UAP0					Process add. payments and deductions
000120	COPY	UAL0					Proration and cumulation gross
000130	COPY	UTBS					Save tables for iteration
000140	LPBEG						Begin of iteration
000150	COPY	UTBL					Load saved tables
000160	COPY	UDD0					Process deductions, Benefits
000170	COPY	UTX0					Calculate taxes
000180	COPY	UGRN					Calculate garnishments
000190	COPY	UNA0					Calculate net
000200	COPY	UDNT					Deductions not taken during loop ?
000210	LPEND						End of iteration
000220	ENDIF						to: if non authorized manual check (*)
000230	BLOCK	END					
000240	COPY	UGRR					Garnishment Retroactive
000250	COPY	URR0					Retroactive accounting
000260	COPY	UNN0					Net processing
000270	COPY	UAC0					Month end accruals
000280	COPY	UEND					Final processing

Figure 4.3 Schema U000

The discussion of the Figures 4.2 and 4.3 uses technical terms, such as *rules*, *operations*, and *functions*. Before proceeding with the detailed discussion about schemas, it's important for you to understand the definitions of these three terms. We will first look at the definitions of schemas and subschemas and then examine the definitions of functions, rules, and operations.

4.2.1 Schemas

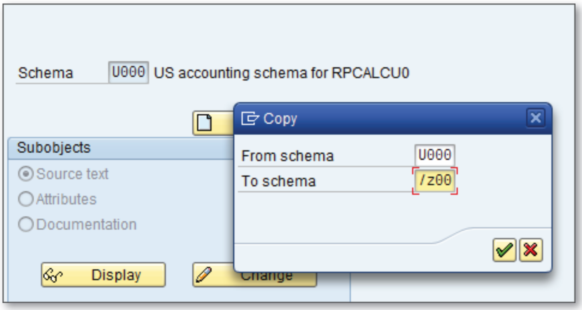
The schema, or main schema, consists of many subschemas. For example, U.S. schema U000 contains subschemas UAP0, UAL0, and UTX0, among others. Each schema has a four-character name. As noted earlier, the *U is* the U.S. country code. Schemas that start with the letter *X* are international and can be used for many countries. The statements and code shown in Figure 4.3 are for the main schema U000. Table 4.1 explains the meanings of the columns shown in Figure 4.3.

Column in Schema Screen	Explanation
Line Numbers	Are sequential. Note that you can insert or delete lines.
Func.	Stands for functions, which run when you have an ABAP code in the background and carry out specific instructions, such as reading infotypes.
Par1, Par2, Par3, Par4	Functions have parameters and can behave differently depending on the value of the parameters. There can be up to four parameters, as shown in Figure 4.5.
D	When you have an asterisk (*) in this column, it means the line is commented and will not be executed. You use commenting when modifying schemas.
Text	Free-form text is used to describe the purpose of the schema line.

Table 4.1 Schema Editor Columns

4.2.2 Subschemas

Subschemas form part of the main schema, and each subschema performs specific functions. For example, UNA0 is used for net calculations, while UTX0 corresponds to tax calculations. In Figure 4.4, you can see that UIN0, UBD0, and UPR0 are subschemas. You use the COPY function to copy subschemas into the main schema. At any given time, there will be one main schema and many subschemas; however, the term *schema* can be generically used for both the main and subschemas.



Schema U000 US accounting schema for RPCALCU0

Subobjects
● Source text
○ Attributes
○ Documentation

From schema U000
To schema /Z00

Figure 4.4 Copying a Delivered Schema

Functions

Functions are typically used to read infotypes in schemas and subschemas. They also perform other tasks, such as printing and rule execution. For example, see the COPY and BLOCK functions in Figure 4.4.

Rules

Rules are used to calculate and manipulate wage types. You'll learn more about rules as you explore the subschemas in this discussion. You'll need to expand the schema or subschema further to make the rules visible. Just like schemas, rule names can start with a *U* (for the United States) or an *X* (for international). Later in this chapter, you will create a custom rule following custom naming specifications.

Operations

Rules use operations for mathematical and logical processing. For example, operations can perform tasks such as multiplication, division, percentage calculation, wage type writing, and so on. You will learn more about operations later in Section 4.5.

SAP includes schemas, subschemas, rules, and operations, and you should copy and modify the schemas and subschemas as needed. All of these objects have editors that are accessed via SAP transactions such as PE01, PE02, and so on. (These are listed in Table 4.2.) When you edit these objects using the editors, SAP creates a transport so you can send the objects across landscape development to testing to production. Configuration and system maintenance teams work with these objects.

Editor Transaction	Objects to Edit
PE01	Schemas and subschemas
PE02	Rules
PE03	Features
PE04	Functions and operations

Table 4.2 Editor Transactions for Schemas and Their Children

Tip
Don't forget the golden rule of SAP configuration: *Never modify SAP-delivered objects*. Instead, always copy and create new ones in a customer name space. SAP typically displays warning messages if you try to create objects in SAP's namespace rather than using customer-specific names, such as those starting with Z.

To understand schema U000, you will also need to learn about its many subschemas. The next few sections explore those subschemas in depth.

4.2.3 U.S. Subschemas

In Figure 4.3, you saw that subschemas form part of the main U.S. schema, U000. Table 4.3 presents a simple list of all applicable U.S. subschemas and a brief description of their functionality.

Subschema Name	Description of Subschema Functionality
U001	ADP schema for RPCALCU0. For gross to net with ADP (third-party payroll processing)
U500	Used during third-party remittance processing for taxes and benefits
UAC0	Personnel calculation schema for month-end accruals
UAL0	Gross calculation
UAP0	Reading in further pay and deductions
UBD0	Basic data

Table 4.3 U.S. Subschema List

Subschema Name	Description of Subschema Functionality
UBE1	Process benefits (first time); health and insurance plans
UBE2	Process benefits (second time); savings and flex spending plans
UBEN	Process benefit infotypes
UCLM	Standard schema used during claims processing (overpayments)
UDBS	Deduction-based calculation
UDD0	Gross calculation
UDP0	Processing of deductions/donations and their balances and totals
UEND	Final processing
UGRN	Garnishments
UGRR	Garnishments (retro calculation)
UIN0	Initialization of accounting
ULK0	Transfers payroll results from Table 558A/5U8A (U.S.-specific) to the payroll results tables. This is used if the payroll system is going live in the middle of the year. (You will learn more about this schema in Chapter 9.)
ULK9	Transfers payroll accounts from Table 5U8C (U.S.-specific) to the payroll results tables. This is also used if the payroll system is going live in the middle of the year
UMC0	Processes manual checks and is used during off-cycle payroll. We will refer to this schema for year-end off-cycle payrolls in Chapter 8.
UNA0	Net calculation
UNN0	Net payments/deductions and transfers
UT00	Processing of time data for payroll accounting
UTX0	U.S. tax processing

Table 4.3 U.S. Subschema List (Cont.)

You will need to copy and modify some of the schemas listed in Table 4.3 so you can insert new customized rules of your own. There is no fixed guidance about commonly modified subschemas because they depend entirely on your requirements. However, if you decide to add or change a rule in any of the subschemas, you will have to copy and modify the subschema as necessary. The following sections closely examine each subschema to help you learn about their individual functionality. They won't be analyzed line by line; however, important features of each will be highlighted. In SAP configuration (IMG), these subschemas appear at different places as they are relevant. For example, the tax subschema appears under U.S. tax configuration, while the month-end accrual schema appears under month-end accrual configuration. However, you can easily access any subschema using the PE01 editor transaction. As you are reading any subschema through its main schema, it is easy to understand the flow of the logic and how the control is passed from one subschema to another. After you start the main schema in the editor, DOUBLE-CLICK each subschema in the editor to open it.

In each schema, functions, rules, and tables are the three most important dimensions you need to know. Please refer to Appendix A for detailed U000 schema lines, as discussed here.

UIN0: Initialization of Payroll

This schema manages the status of a payroll control record. You can comment the function CHECK if you want to bypass the check for the control record. There is one main feature of this schema. Function CHECK (parameter ABR) checks the payroll control record status. During the testing stage of the project, you might want to comment this line in the schema so that, regardless of the payroll control record's status, you will be able to test-run your payroll. In a production payroll system, you should never comment this line.

UBD0: Basic Data Processing

After the initial checks on the control record are carried out, this schema checks for the existence of basic infotypes from the employee master data. Many of the infotypes discussed in Chapter 2 are processed in the schema. The following are the main features of this schema:

- Function WPBP (work place/basic pay) manages any splits in an employee's Infotype 0001, 0007, 0008, or 0027. For example, if an employee joins the organization in the middle of a pay period, or if an employee takes a new position in the middle of a pay period, you will see the impact of this function. In addition, if an employee's work schedule changes in the middle of a pay period, this function creates table WPBP in payroll results.

- Functions P0002, P0006, P0207, and P0014 read the respective infotypes. For example, if Infotype 0207 (residence tax area) is missing from your employee master data, the schema will cause an error when it reaches this subschema. Also, if you are interested in finding out the "internal" processing of data from Infotype 0014, you can drill down into the schema here. (*Drill-down* is a general term meaning to explode the schema to subschemas and rules.) Various employee master data or transaction data infotypes discussed in Chapter 2 are processed in the schema using these functions. In Appendix A, similar functions refer to functions P2010, P0221, and so on, and they process the respective infotypes.
- Rule UW14 is used to process data from Infotype 0014 with function P0014. The wage types from Infotype 0014 form input and output to this rule. For example, a recurring deduction for the United Way is maintained in Infotype 0014 and is processed by this rule.
- Several tables are relevant for subschemas UBD0:
  - **WPBP** is created by function WPBP and is visible in payroll results.
  - **IT (input table)** contains wage types with their amounts, numbers, and rate information.
  - **NAME** is created from Infotype 0002.
  - **ADR** is created from Infotype 0006.
  - **TAXR** contains the residence tax authority and is created from Infotype 0207.

Tip

At this point, we recommend that you revisit and review SAP's documentation on payroll tables and their use.

UT00: Gross Compensation and Time

UT00 is probably one of the most important subschemas. This subschema deals with time management, gross compensation, and rate calculations. It is appropriate at this point to remind you about the integration between the Time Management and Payroll subcomponents. You might even be wondering whether your implementation uses time evaluation. If it does, then this subschema reads time evaluation results (B2 time clusters). Alternatively, if your system is *not* using time evaluation, then Infotype 0007 and 0008 will govern the calculations.

This schema handles many important functions (such as ZLIT) where the time management and payroll processes come together. The schema also has key rules such as X010 and X012 for valuation of wage types. In addition, the schema processes tables, discussed in the following:

- Function PARTT acts if an employee joins in the middle of the month or in situations where the schema has to perform partial-month calculations. When a partial-month calculation is performed, a table PARX is generated, as listed later in this section.
- Function P2003 processes the substitution infotype. Since Infotype 2003 falls under the Time Management component in SAP systems, it is naturally processed in this time subschema.
- Function P2010 relates to the employee remuneration infotype. Many SAP customers use Infotype 2010 either to load time data from external time systems or to manually enter time data in the absence of any time-clocking systems. Therefore, if you need to write any rules related to data in Infotype 2010, UT00 is the subschema to use.
- Function DAYPR checks the time processing with or without clock times.
- Function ZLIT brings the hours from time management (either time evaluation or normal working hours) together with the dollar rates from payroll. For example, if an employee needs to get paid overtime at 1.5 times the normal rate, the schema needs to know the number of hours and the employee's hourly rate. The ZLIT function brings the two together: hours and rate.
- Rule X013 performs the valuation. For example, if an employee is salaried, you may need to calculate that employee's derived hourly rate for partial-month calculations. In Chapter 3 you saw wage types /001 and /002. They are the valuation technical wage types that are generated in this subschema.
- Rule X020 collects the wage types using operation ADDCU. See Appendix F for a list and description of this and other operations.
- Several tables are relevant for subschema UT00.
    - PARX is generated if the function PARTT finds a partial period processing case during payroll processing. For example, this would be the case if a pay period

is from August 8 through August 22, and an employee joined the company on August 14.

- C1 is generated when an employee has worked across different cost centers in a single pay period, which might happen if the employee travels between branches of a company. In a normal scenario, if an employee is salaried and always works in the home cost center, you won't see the impact on this table.
- ZL contains time wage types. Time wage types in this table contain hours, but not amounts. As discussed earlier for the ZLIT function, hours from the ZL table are used with rates from the IT table to arrive at amounts.
- IT (input table) flows once again through the schema, building additional wage types as the table is passed through more subschema and rules. Therefore, the IT table you see in this schema will look different for contents and wage types compared to schema UBD0. As mentioned earlier, function ZLIT brings the ZL and IT tables together.

UMC0: Non-Authorized Check

In payroll processing, you can have two major payroll types: *normal* and *off-cycle*. In Chapter 8, we'll look at the year-end adjustment payrolls using Infotype 0221. Since this subschema addresses off-cycle payroll processing, the logic from this subschema will again repeat for normal payroll processing in schema UDD0. The next section describes the functions, rules, and tables in this schema.

- Function P0221 reads Infotype 0221, which is normally used in year-end adjustments and manual checks. For more examples using Infotype 0221, see Chapter 7.
- Rule UNAM processes Infotype 0221 wage types.
- Rules X023, X024, and X025 are related to gross calculations. Please note that the respective processing classes 20, 41, and 4 (P20, P41, and P04) for the wage types are used in these rules.
- The rules with processing class 66 (P66) are related to the wage types that have a goal/deduction scenario. Infotypes 0014 and 0015 carry these wage types as discussed in Chapter 2 and Chapter 3 (related to master data and wage types), respectively. These rules generate the payroll results wage types associated with deduction as well as balance.
- Rules UD11 and UD21 are related to retroactive calculations. During retroactive calculations, the payroll schema needs to keep track of intermediate wage types.

These wage types are referred to as *inflow wage types*. In Appendix C you will find wage types /X02 and /Z02 at the end of the list of technical wage types.

- Several tables are relevant for UMC0.
  - IT (input table) continues to build further with wage types getting updates from the rules in the schema. New wage types are added to the IT table (for example, goals and deduction processing). Throughout schema processing, the IT table is updated with additional wage types as they get processed.
  - ARRRS (arrears) is applied if the payroll processing does not have enough money in the current pay period, depending on the configuration of the wage type, which means that in the next pay period, if the payroll has enough money, this deduction is taken out of that payroll. If any of the wage types go into arrears processing, the payroll result tables will have an ARRRS table generated in this schema.
  - DDNTK (deductions not taken) contains the deduction wage types that the payroll could not deduct in the current pay processing. This situation occurs as a result of not having sufficient earnings for an employee. Depending on your configuration, the wage type can be in both the arrears and deductions not taken tables (tables ARRRS and DDNTK).

UTX0: Tax Processing

If you refer back to the payroll concept diagram (see Figure 1.2 in Chapter 1), you might recall that SAP uses the Business Software Inc. (BSI) Tax Factory to calculate taxes for U.S. payroll. Almost everyone is familiar with the definitions of *gross payroll* and *net payroll*. If you are running the net payroll, then the taxes are calculated within the payroll schema by this subschema. This subschema processes taxes and creates tax wage types by different tax authorities. Chapter 8 discusses tax processing and tax authorities at greater length. As with previous schemas, the functions and rules are discussed as follows:

- Function UPAR1 with parameter BSI controls the BSI version. (At the time of this writing, the BSI product is at version 9.0.) BSI Tax Factory is a third-party product that is integrated with SAP US Payroll and takes care of federal, state, and local tax calculations. BSI handles tax policies, tax rates, and any annual changes. In the same way that SAP sends out support packs on a regular basis, BSI sends tax update bulletins (TUBS) to their customers on a regular basis.

- Function USTAX passes control to BSI and brings back the tax wage types from BSI. The tax wage types (/401, /402, /403, etc.) discussed in Chapter 3 are generated in BSI and are sent to the payroll schema when processing this function.
- Rule UPTX separates the tax amounts for each of the tax authorities. Recall that, as discussed in Chapter 2, Infotypes 0207 and 0208 have multiple tax authorities and levels (federal, state, and local). You will learn about rules and review typical changes to rule UPTX later in this chapter.
- Several tables are relevant for schema UTX0.
  - IT (input table) is visible in the schema log and is not visible in employee payroll results.
  - RT (results table) is eventually visible in an employee's payroll history; the schema starts building them in and around subschema UTX0.
  - TCRT is used for cumulative tax results.
  - TAXR contains tax authorities.
  - V0 tables are also called split tables, as they contain tax authorities.

UAP0: Process Additional Payments/Deductions

As discussed in Chapters 2 and 3, the payroll process has different deduction and earning types. These earnings and deductions are fed to the payroll process through infotypes in employee data. In Chapter 5, you will learn about the integration between the US Benefits and US Payroll components. This subschema has another subschema (UBE1) nested within it, which processes the U.S. benefits for health and insurance plans.

Later in this chapter is a section on modifying schemas. There you will see the steps necessary to copy and modify a schema in that section to, for example, modify subschema UAP0. The features, rules, and tables in this schema are as follows:

- Functions P0014 and P0015 process Infotypes 0014 and 0015. Also note that rules U011 and U015 are used in the processing.
- Function P0267 handles off-cycle/bonus payments with rule U012.
- Function COPY UBE1 copies subschema UBE1 for benefits processing. The statements between BLOCK BEG and BLOCK END correspond to subschema UBE1.

- Functions P0167 and P0168 process health plans (Infotype 0167) and insurance plans (Infotype 0168), respectively. Therefore, if you need to drill down and examine the benefits amount or calculations, this is the place to do so.
- Rule ZRU1 is a custom rule created for this schema. This rule is examined in depth later, in Section 4.3, during the discussion about customized rules.
- Several tables are relevant for UAP0.
    - IT (input table)
    - V0 split tables for different benefit plans

UAL0: Proration and Cumulation

In Chapter 3, you learned about wage type cumulation and prorated wage type calculations. This schema processes wage type cumulations as controlled by processing class 20.

- Function GEN/8 and rule XPPF are used to generate wage types /801 and /802, respectively. If a wage type needs to be prorated for the pay period (proration is the factoring of a wage type for partial pay period calculations), then this function and rule should be used.
- Rule XPPF generates wage types /801 and /802.
- Rule X023 uses processing class 20 (P20) and the wage types in result tables of payroll.
- IT (input tables) are most relevant for UAL0.

UDD0: Process Deduction and Benefits

The UDD0 subschema plays an important role in retroactive accounting. It also handles intermediate wage types during retroactive calculation as the payroll schema runs multiple times, depending on the number of retro periods.

For example, if during payroll period 17, retroactive accounting demands adjustment of period 15, the schema will run for periods 17, 16, and 15. In that situation, subschema UDD0 handles the intermediate flow of wage types. It has two nested subschemas: UBE2 for U.S. benefits and UDP0 for handling deduction goals and totals. Earlier in subschema UAP0, the benefits subschema UBE1 will have processed health and insurance deductions; now subschema UBE2 processes savings, flexible spending, and miscellaneous benefit plan–related deductions. Some logic from schema UMC0 has repeated again. For UMC0, the logic was related to non-authorized manual check off-cycle payroll processing, and now for subschema UDD0, the same logic relates to normal payroll processing. The functions, rules, and tables in this schema are as follows:

- Function COPY UBE2 processes savings plans and flexible spending accounts in benefits through subschema UBE2.
- Function P0170 processes flexible spending account plans.
- Function P0169 processes savings plans (401(k)).
- Function P0377 processes miscellaneous benefit plan deductions. If you need to know more about this or any of the earlier functions, please use Transaction PE04 to access these functions. The same transaction also offers access to documentation for these functions.
- Function COPY UDP0 manages a subschema to process Infotype 0014- and 0015-based deductions with goal amounts wage types.
- Function LIMIT checks if the amounts in certain wage types exceed a limit. This function works in conjunction with arrears processing to ensure that deductions that went into arrears do not exceed the arrears amounts.
- Function PRDNT works with the DDNTK table. Like arrears processing, DDNTK processing has a deeper impact on retroactive accounting calculations.
- Rules UD11 and UD21 are related to retroactive calculations. During retroactive calculations, the payroll schema needs to keep track of intermediate wage types. These wage types are referred to as inflow wage types. In Appendix A3, which lists the technical wage types, you will find wage types /X02 and /Z02 at the end of the list; these are the wage types handled by these rules.
- Rules UD**. Rules using processing class 66 (P66) manage goals and deduction wage types. (Note that these rules changed to P50 in newer versions of SAP HCM.)
- Rules X024 and X025 manage cumulation of wage types using processing class 41 and 04, respectively.
- Several tables are relevant for subschema UDD0; their descriptions can be found earlier in this section.
  - IT (input table)
  - ARRRS (See the discussion of schema UMC0)
  - DDNTK (See the discussion of schema UMC0)

UGRN: Garnishments Calculation

Chapter 7 focuses on garnishments, covering this topic in depth. When you use the discussion points from that chapter and run your tests in the payroll, subschema UGRN forms an important part of the equation. Various functions and rules related to garnishments are discussed here. Also note that the tables generated in this schema are different from those generated in other schemas and are naturally oriented to garnishments.

- Function IF with the parameter GREX checks for active garnishments. Active garnishments have their status set via Infotype 0194. Garnishments that are inactive and are in released status aren't processed in the schema.
- Function UGARN performs calculations of garnishments.
- Rule UGRT with processing class 59 (P59) processes all earnings that have garnishability. This rule is read by function PRT. (Just as PIT helps rules read IT tables, PRT helps rules read RT tables.)
- Rule UGDN calculates disposable net income, which is discussed in Chapter 7.
- Several tables are relevant for UGRN.
  - GRDOC contains garnishment documents.
  - GRREC contains the garnishment record.
  - GRORD contains the garnishment order.
  - IT tables are generated like with all other subschemas.

In Chapter 7, we will see the contents of these tables in subschema UGRN.

UNA0: Calculate Net Pay

As we get closer to the end of schema U000, the net pay is calculated with schema UNA0. If you need to write rules before final result tables are written, this is the subschema to use. You will mostly be concerned with the IT and RT tables in this subschema, which by now have completed the logic and calculation portions of the schema.

UNN0: Net Processing Bank Transfer

The payroll processing isn't complete until it processes check payments or bank transfers, as chosen by the employee in Infotype 0009, which is discussed in Chapter 2. Schema UNN0 uses the infotypes to create bank transfers or check payment tables.

- Function P0011 is used if your implementation is using Infotype 0011 (external bank transfer).
- Function P0009 processes Infotype 0009 bank/check data using details such as the bank routing number, account number, and so on.
- The schema uses function P9ZNC if any of the employees have to receive checks with an amount of zero. It is also recommended that you run the wage type reporter or payroll journal for such cases.
- Several tables are relevant for schema UNN0.
  - BT (bank table) for bank transfer/checks printing
  - Results tables

UEND: Final Processing

Final processing with schema UEND serves a single, very important purpose: presenting you with the payroll results table that will be available in the payroll clusters. These tables will be used for all subsequent processing, such as finance posting, accounts payable posting, checks processing, tax processing, and payroll reporting in general (as seen in Chapter 1, Figure 1.1). These tables contain an individual employee's payroll results for each pay period. When running the schema using the log option, this is where you drill down to check the results with the following tables:

- RT
- CRT (cumulative results table), which contains MTD and YTD (month-to-date and year-to-date) accumulations

Now that you have an overview of subschema functionality, you're ready to learn how to modify schemas using simple examples.

4.3 Why, When, and How to Change the U.S. Schema

By now you have seen many subschemas, and you might feel that they will meet all of your payroll processing needs. If that's the case, you're probably asking *Why do I need to modify the included schema and subschemas?* Table 4.4 lists some practical US Payroll requirements and solutions to help answer this question.

Your Requirement	Suggested Solution	Impact on Schema Modification
Part of your benefits are outsourced, and you don't want SAP Payroll to calculate those deductions.	Modify subschemas UBE1 and UBE2.	Copy and modify subschemas UBE1 and UBE2.
You have a customized formula to calculate your union deduction.	Build the formula in a rule.	Copy the appropriate delivered subschema and insert the new rule in it.
You need to generate a wage type for certain groups of employees, using a specific base amount for calculation.	Create a new rule that checks the employee grouping and creates a wage type. This rule needs to be inserted in the appropriate subschema.	Copy the appropriate delivered subschema and insert the new rule in it.
You calculate per-pay-period payroll service costs based on an employee's base salary and post them to a General Ledger (GL) account.	Create a new rule that generates the wage type and is mapped to a particular GL account for posting.	Copy the appropriate delivered subschema and insert the new rule in it.
You have not yet installed BSI and still want to continue other testing.	Comment out the USTAX function in schema UTX0.	Copy schema UTX0 to schema ZTX0 and comment out the line.
You have to post the tax wage types for different tax authorities to different GL accounts.	Modify the UPTX rule.	Copy schema UTX0 to schema ZTX0, comment out the UPTX line, and then insert a new line for the ZPTX rule.

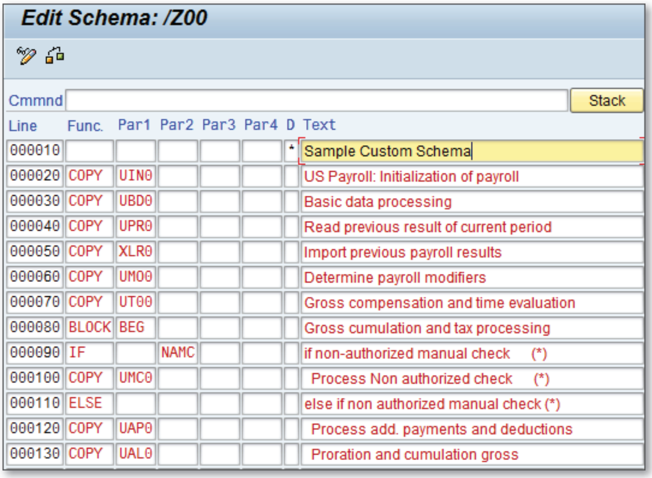
Table 4.4 Sample Payroll Requirements that Drive Schema Modifications

These are just a few of the many requirements that can drive rule modifications and therefore subschemas. Each industry can have its own requirements and can sometimes demand very unique situations that can be handled through custom rules and custom subschemas. Next, you will learn how to modify schemas using the Schema Editor.

4.3.1 Copy and Modify Schema U000 using the Schema Editor

Using a step-by-step approach, we'll use the Schema Editor to copy and modify Transaction PE01.

1. **Open the Schema Editor and copy schema U000.**
   To start working with the schema, first copy the U000 schema and create your own version named /Z00. Using Transaction PE01, create schema /Z00, as shown earlier in Figure 4.4. When creating new schemas, it is best to keep the naming convention in mind. Do not use SAP's name space; it is safer to use Z or / (slash) as the starting character.
2. **Modify new copies of schema /Z00 using the editor.**
   After you copy and create the /Z00 schema, select the Schema Editor's Change option to go to the screen shown in Figure 4.5.



Edit Schema: /Z00

Cmmnd

Stack

Line	Func.	Par1	Par2	Par3	Par4	D	Text
000010						*	Sample Custom Schema
000020	COPY	UIN0					US Payroll: Initialization of payroll
000030	COPY	UBD0					Basic data processing
000040	COPY	UPR0					Read previous result of current period
000050	COPY	XLR0					Import previous payroll results
000060	COPY	UMO0					Determine payroll modifiers
000070	COPY	UT00					Gross compensation and time evaluation
000080	BLOCK	BEG					Gross cumulation and tax processing
000090	IF		NAMC				if non-authorized manual check (*)
000100	COPY	UMC0					Process Non authorized check (*)
000110	ELSE						else if non authorized manual check (*)
000120	COPY	UAP0					Process add. payments and deductions
000130	COPY	UAL0					Proration and cumulation gross

Figure 4.5 Modifying the New Schema /Z00

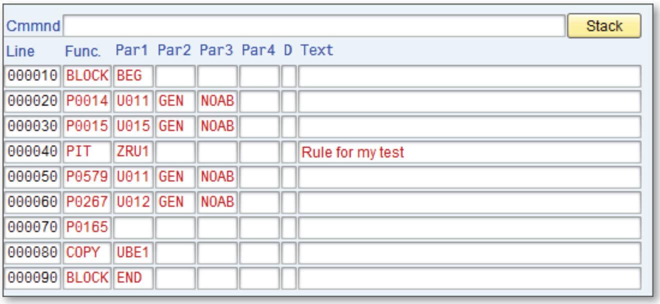
You should follow this process to copy and modify any subschemas. The following changes were made to schema /Z00, as shown in Figure 4.5:

- **Line 00010**
  Added a comment line at the beginning of the schema.
- **Line 00140**
  Commented out the line for standard schema UAP0, which means that UAP0 isn't processed by the /Z000 main schema. It is a good practice to keep the

original line commented out rather than deleting it. This eases maintenance, and it is also helpful to know what the schema looked like before and after changes were made.

- **Line 00150**
  A new line is added, containing the copied and modified subschema ZAP0. To add a new line in the Schema Editor, place the cursor at a line number and insert the letter "*i*." The Schema Editor adds a new blank line at the cursor's position. Similarly, to delete a line, place the cursor at a line number and enter the letter "*d*."

3. **Drill down to subschema ZAP0 from the main schema /Z00.**
   Next, place the cursor at line 00150 and double-click to expand subschema ZAP0. Figure 4.6 shows the subschema ZAP0 in an exploded, or detailed, fashion. Add a new rule called ZRU1 at line number 000040. Don't worry about the rules and why ZRU1 is in the subschema. At this stage, just focus on copying and modifying the U.S. schema to your own version /Z00. You will learn the steps to create rules later in this chapter.



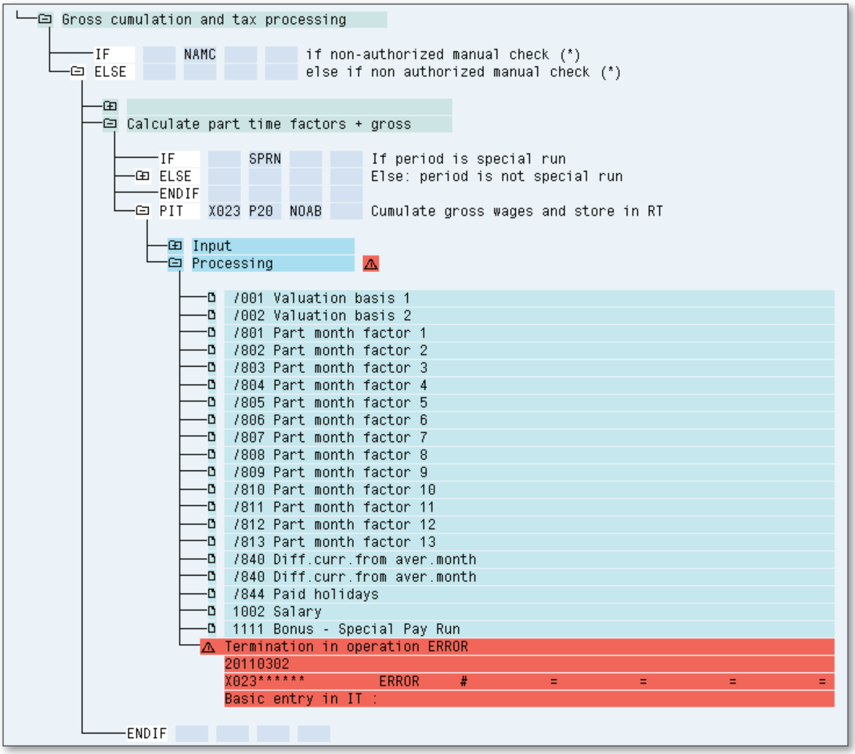
Line	Func.	Par1	Par2	Par3	Par4	D	Text
000010	BLOCK	BEG					
000020	P0014	U011	GEN	NOAB			
000030	P0015	U015	GEN	NOAB			
000040	PIT	ZRU1					Rule for my test
000050	P0579	U011	GEN	NOAB			
000060	P0267	U012	GEN	NOAB			
000070	P0165						
000080	COPY	UBE1					
000090	BLOCK	END					

Figure 4.6 Explosion of Subschema ZAP0

After you have made changes to any of the subschemas, it is a good idea to generate the main schema. This helps you catch any syntax errors before running the schema through the payroll driver. You will find the GENERATE button in the Schema Editor screen itself. You are now ready for a trial run of the schema. As discussed at the beginning of this chapter, you will need the US Payroll driver RPCALCU0 to run the schema.

4.3.2 Running an Error-Free Schema

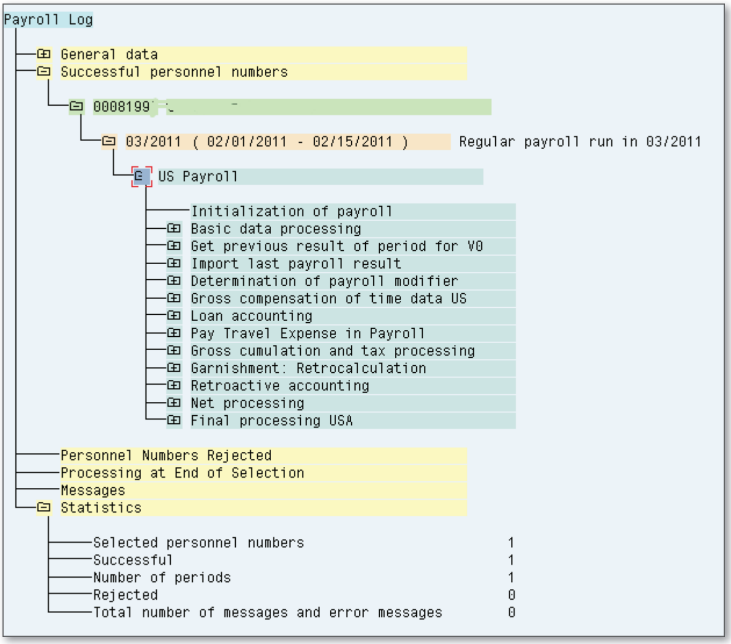
Using the US Payroll driver RPCALCU0, run schema /Z00. Make sure to turn the log button ON for testing, and as shown in Figure 4.7, an error will be visible. Note that we're testing the schema with just one employee. You would normally schedule an RPCALCU0 batch run for all employees. If your schema encounters a runtime error, the error will be clearly visible on your SAP screen, as shown in Figure 4.7. You need to analyze and fix the error and then run the driver. In this example, there seems to be an error with wage type 0750 with an operation in the rule. Note the error message (TERMINATION IN OPERATION ERROR) that identifies the error. This also means you need to focus on checking the specific rule where the termination has occurred for wage type 1111.



```
Gross cumulation and tax processing
  IF     NAMC        if non-authorized manual check (*)
  ELSE               else if non authorized manual check (*)

    Calculate part time factors + gross
      IF      SPRN        If period is special run
      ELSE                Else: period is not special run
      ENDIF
      PIT     X023 P20 NOAB    Cumulate gross wages and store in RT
        Input
        Processing  ⚠
          /001 Valuation basis 1
          /002 Valuation basis 2
          /801 Part month factor 1
          /802 Part month factor 2
          /803 Part month factor 3
          /804 Part month factor 4
          /805 Part month factor 5
          /806 Part month factor 6
          /807 Part month factor 7
          /808 Part month factor 8
          /809 Part month factor 9
          /810 Part month factor 10
          /811 Part month factor 11
          /812 Part month factor 12
          /813 Part month factor 13
          /840 Diff.curr.from aver.month
          /840 Diff.curr.from aver.month
          /844 Paid holidays
          1002 Salary
          1111 Bonus - Special Pay Run
        ⚠ Termination in operation ERROR
          20110302
          X023******          ERROR     #          =          =          =          =
          Basic entry in IT :
    ENDIF
```

Figure 4.7 Error While Running the Schema

If the schema runs without errors, you will also see statistics at the end, listing the number of employees included in the run, as shown in Figure 4.8.



```
Payroll Log
 ├─⊞ General data
 ├─⊟ Successful personnel numbers
 │   └─⊟ 0008199
 │       └─⊟ 03/2011 ( 02/01/2011 - 02/15/2011 )    Regular payroll run in 03/2011
 │           └─⊟ US Payroll
 │               ├──── Initialization of payroll
 │               ├─⊞ Basic data processing
 │               ├─⊞ Get previous result of period for V0
 │               ├─⊞ Import last payroll result
 │               ├─⊞ Determination of payroll modifier
 │               ├─⊞ Gross compensation of time data US
 │               ├─⊞ Loan accounting
 │               ├─⊞ Pay Travel Expense in Payroll
 │               ├─⊞ Gross cumulation and tax processing
 │               ├─⊞ Garnishment: Retrocalculation
 │               ├─⊞ Retroactive accounting
 │               ├─⊞ Net processing
 │               └─⊞ Final processing USA
 ├──Personnel Numbers Rejected
 ├──Processing at End of Selection
 ├──Messages
 └─⊟ Statistics
     ├──Selected personnel numbers                        1
     ├──Successful                                        1
     ├──Number of periods                                 1
     ├──Rejected                                          0
     └──Total number of messages and error messages       0
```

Figure 4.8 An Error-Free Schema Run

The next obvious question is *What are the typical conditions that will cause the schema to error out?* Table 4.5 is sample list of errors and possible resolutions, but it is by no means a complete list, as every situation can have different error types.

Error Category	Example of Error Condition	Resolution
Employee master data	Missing employee master data infotypes.	If any of the mandatory infotypes, such as Infotype 0008, 0207, or 0009 are missing, the schema will "error out."
Rules and operations	You might have added a new custom rule to the schema that has operation errors.	Rules normally have errors related to wage types and/or operations.

Table 4.5 Sample Error Conditions During a Schema Run

Error Category	Example of Error Condition	Resolution
Wage type	Missing specifications of wage type processing class in a rule.	Many rules use processing class specification values for wage types as decision criteria. If the wage type has a missing value for a particular specification, the schema will error out.
Employee master data	Infotype 0003 has improper dates.	The employee's master data dates and payroll run dates do not match.
Garnishment data	Missing master data for garnishment.	Garnishment Infotype 0195 master data can cause errors in the payroll schema.
Rules and operations	In the rules editor, you need to follow a certain indentation convention for operations, which you will learn more about in the next section when writing rules. If the operation in a rule is coded at the wrong place, then the schema will have an error.	The operation does not give an error, because it doesn't perform the desired tasks specified by the rule.

Table 4.5 Sample Error Conditions During a Schema Run (Cont.)

After all errors are cleared and you have performed an error-free run of the schema (shown in Figure 4.8), you're ready to drill down into (explode) the schema. The drill-down has a simple structure, and although it looks intimidating at first, it is not difficult to work with. Before we start the drill-down of the schema, Table 4.6 lists the possible tables you will see during and after the payroll processing. Only tables relevant to the discussion are listed.

Note

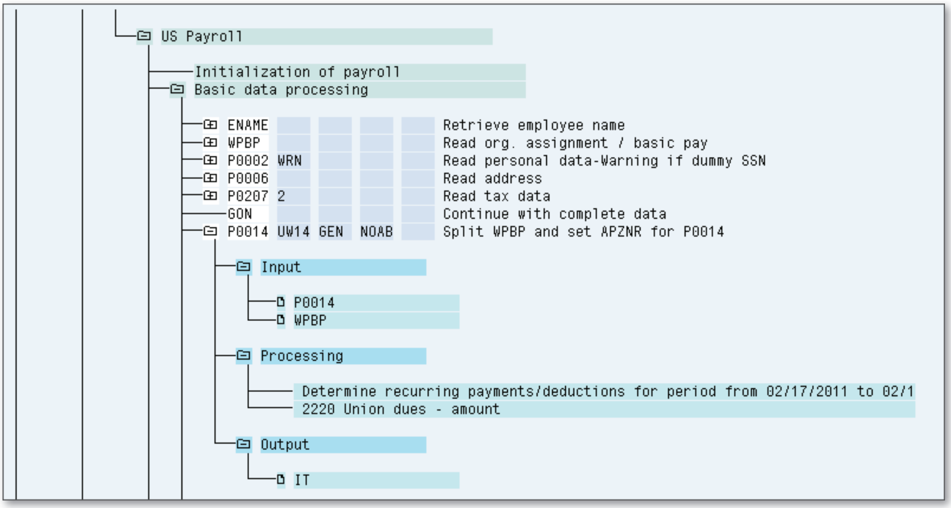
If you require additional information, or an expanded list, visit SAP's standard documentation (*http://help.sap.com*) for a discussion of all available tables. You will find this documentation under generic (not country-specific) payroll components (PY-XX-BS).

Table	Description of Table	Used During Processing	Available After Processing
IT	Input table. Used during processing and passes data to the RT tables. Your rules will typically write wage types to these tables.	✔	
RT	Results table. Final table where payroll results are stored.		✔
CRT	Cumulative results table. Contains month-to-date and year-to-date details.		✔
ARRRS	Contains the deductions that are carried over to the next payroll period. See UMC0: Non-Authorized Check in Section 4.2.3 for additional information.		✔
DDNTK	Contains the deductions not taken during the current pay period. See UMC0: Non-Authorized Check in Section 4.2.3 for additional information.		✔
ACCR	Month-end accrual tables, discussed further in Chapter 9.		✔
WPBP	The work place/basic pay (WPBP) table is created due to changes to Infotypes 0001, 0027, 0007, and 0008. See UBD0: Basic Data Processing in Section 4.2.3 for additional information.		✔

Table 4.6 Payroll Tables

Now let's follow the drill-down steps to check the table contents as well as the processing of the schema.

1. **Drill down into the schema and get to the exact subschema or rule location.**
   Figure 4.9 shows the schema at function P0014 and rule UW14. Explode the schema in the same way you would click and open folders in Microsoft Windows Explorer.

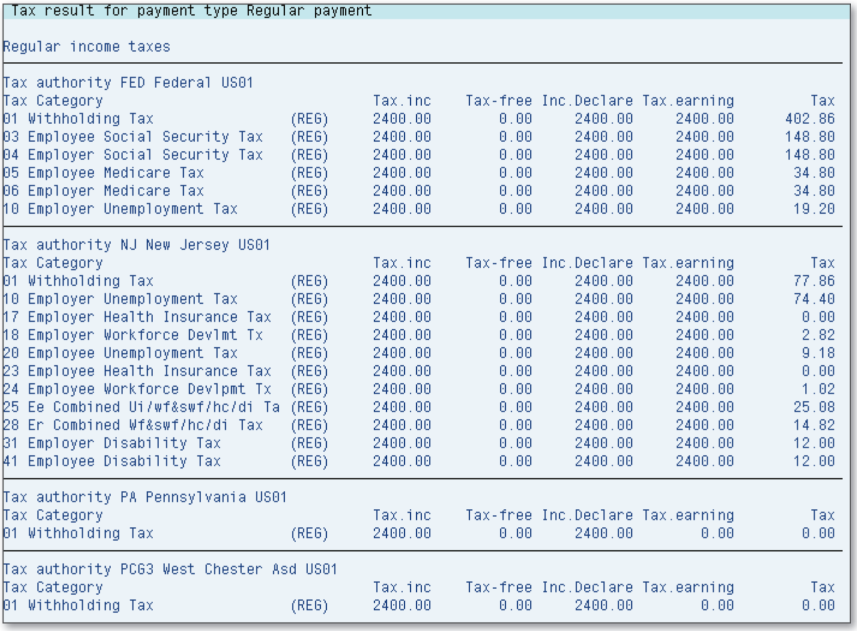


```
US Payroll
  Initialization of payroll
  Basic data processing
    ENAME                          Retrieve employee name
    WPBP                           Read org. assignment / basic pay
    P0002  WRN                     Read personal data-Warning if dummy SSN
    P0006                          Read address
    P0207  2                       Read tax data
    GON                            Continue with complete data
    P0014  UW14  GEN   NOAB        Split WPBP and set APZNR for P0014
      Input
        P0014
        WPBP
      Processing
        Determine recurring payments/deductions for period from 02/17/2011 to 02/1
        2220 Union dues - amount
      Output
        IT
```

Figure 4.9 Drill-Down Schema and Rule

2. **Click on INPUT, OUTPUT, or PROCESSING for the rule.**
   You can click on any of the tables visible under OUTPUT. In many rules and functions, you will see the IT tables, but as we have seen in subschemas before, you can also check other tables such as WPBP, RT, ARRRS, and others.

3. **Drill down to get into the specialty functions, such as USTAX processing.**
   In addition to normal rules and standard functions, it also helps to drill down into special areas such as tax processing. If you drill down at the USTAX function in the tax processing subschema (as shown in Figure 4.10), you will be able to see detailed tax calculations for each tax authority that is processed. Figure 4.10 shows the federal tax authority-related calculations within the USTAX function. The figures are for reference only.



```
 Tax result for payment type Regular payment

Regular income taxes
____________________________________________________________________________________________________
Tax authority FED Federal US01
Tax Category                                 Tax.inc    Tax-free Inc.Declare Tax.earning         Tax
01 Withholding Tax                 (REG)     2400.00        0.00     2400.00     2400.00      402.86
03 Employee Social Security Tax    (REG)     2400.00        0.00     2400.00     2400.00      148.80
04 Employer Social Security Tax    (REG)     2400.00        0.00     2400.00     2400.00      148.80
05 Employee Medicare Tax           (REG)     2400.00        0.00     2400.00     2400.00       34.80
06 Employer Medicare Tax           (REG)     2400.00        0.00     2400.00     2400.00       34.80
10 Employer Unemployment Tax       (REG)     2400.00        0.00     2400.00     2400.00       19.20
____________________________________________________________________________________________________
Tax authority NJ New Jersey US01
Tax Category                                 Tax.inc    Tax-free Inc.Declare Tax.earning         Tax
01 Withholding Tax                 (REG)     2400.00        0.00     2400.00     2400.00       77.86
10 Employer Unemployment Tax       (REG)     2400.00        0.00     2400.00     2400.00       74.40
17 Employer Health Insurance Tax   (REG)     2400.00        0.00     2400.00     2400.00        0.00
18 Employer Workforce Devlmt Tx    (REG)     2400.00        0.00     2400.00     2400.00        2.82
20 Employee Unemployment Tax       (REG)     2400.00        0.00     2400.00     2400.00        9.18
23 Employee Health Insurance Tax   (REG)     2400.00        0.00     2400.00     2400.00        0.00
24 Employee Workforce Devlpmt Tx   (REG)     2400.00        0.00     2400.00     2400.00        1.02
25 Ee Combined Ui/wf&swf/hc/di Ta  (REG)     2400.00        0.00     2400.00     2400.00       25.08
28 Er Combined Wf&swf/hc/di Tax    (REG)     2400.00        0.00     2400.00     2400.00       14.82
31 Employer Disability Tax         (REG)     2400.00        0.00     2400.00     2400.00       12.00
41 Employee Disability Tax         (REG)     2400.00        0.00     2400.00     2400.00       12.00
____________________________________________________________________________________________________
Tax authority PA Pennsylvania US01
Tax Category                                 Tax.inc    Tax-free Inc.Declare Tax.earning         Tax
01 Withholding Tax                 (REG)     2400.00        0.00     2400.00        0.00        0.00
____________________________________________________________________________________________________
Tax authority PCG3 West Chester Asd US01
Tax Category                                 Tax.inc    Tax-free Inc.Declare Tax.earning         Tax
01 Withholding Tax                 (REG)     2400.00        0.00     2400.00        0.00        0.00
```

Figure 4.10 Drill Down to the USTAX Function

Schemas and rules—the "heart and soul" of SAP payroll—offer a very powerful and flexible framework. This framework helps users and implementation teams configure the rules, organization-specific calculations, and policies in SAP payroll. This section should have provided you the details of U.S. schema and subschema. SAP has provided specific schemas for public sector and non-profit organizations. In the next sections, we will see the overview of these schemas. The overall framework and philosophy does not change, but you will have additional or different subschemas and functions in these schemas.

4.4 Overview of the U.S. Public Sector Schema Functionality

Since the U.S. public sector has its own unique requirements, SAP has provided additional functionality through a U.S. public sector schema, titled USPS (U.S. public sector). The schema, and its associated functionality, is visible when you implement IS-PS (Industry Solution-Public Sector). This section provides a quick overview of the functionality of this schema. It is not our intention to get into the details of public sector requirements. Public sector-related information is available from multiple resources, including:

- SAP's standard documentation (*http://help.sap.com*)
- The IRS website for non-resident alien taxation (*http://www.irs.gov*)
- Savings bonds (*http://www.treasurydirect.gov*)

The implementations that use IS-PS will give you access to the public sector schema and the documentation associated with it.

4.4.1 NRA: Non-Resident Alien Processing

An *alien* is a person who is not a citizen or national of the United States. A *non-resident alien* (NRA) is someone who is allowed temporary entry in the country for a specific purpose and for a limited amount of time. The IRS has published specific instructions for NRAs. The purpose of this discussion is not to tell you about various forms and instructions regarding NRA processing; instead, the section will highlight SAP's overall functionality for this topic.

Employee master data for NRAs involves the following:

- **Infotype 0094**
  Work permit and residence status that need to be maintained in the employee record
- **Infotype 0048**
  Visa status that needs to be maintained in the employee record
- **Infotype 0556**
  Tax treaty infotype that maintains the treaty groups based on the NRA's home country
- **Infotype 0235**
  Other taxes infotype; can maintain exemptions by the type of taxes based on eligibility and visa status

Payroll schema for NRAs involves the following:

- SAP provides subschemas UPNR and UPPT.
- A statutory reporting schema, 1042S, needs to be performed using Tax Reporter.
- Wage types in wage type group 1042 allow for payroll processing of scholarships.

4.4.2 Savings Bonds Processing

Savings bond deductions and purchases are an important U.S. public sector topic. Savings bond deductions are carried out as per the employee's choice. Typically, the payroll system sends the details via an interface to the U.S. Treasury for savings bond purchases, which are then mailed directly to the employee. Once again, this process starts with the employee master data and ends with interfaces from the payroll system. The savings bond lifecycle steps are listed here:

1. Employee master data for savings bonds is based on the following infotypes:
   - Infotype 0103: Bond purchase amount and deduction amount details
   - Infotype 0104: Bond denomination and name details
2. Payroll schema processing for savings bond deductions:
   - In the U.S. public sector schema, after benefits subschema UBE2, you will find a statement that uses function call P0103 to process savings bond deductions.
   - When an employee decides to leave an organization, you can have a situation where there is a remaining balance. One example could be if an employee has chosen to buy a $100 savings bond with a per-pay-period deduction of $25. After accumulating $50 over the course of two pay periods, the employee decides to leave the company. As a result, the payroll process needs to refund the $50 to the employee. As you delimit the savings bond Infotype 0103, the refund process kicks in, and a refund wage type is created in payroll. This wage type will have no tax impact because the employee already paid tax on this amount before it was taken out as a post-tax deduction.

4.4.3 Public Sector Savings Plans: 403(B), 457(B)

The public and education sectors have different retirement plans than the normal 401(k) plans, which are called the 403(b) and 457(b) plans. These plans are referred by the related tax bills. They are processed in a similar way to 401(k) plans through benefits Infotype 0169. The schema can process them as any other savings plan, but you may also need to write custom rules in schemas to handle the company-specific/implementation-specific investments and accumulation of wage types associated with 403(b) and 457(b) plans.

4.4.4 Employer Benefits and Tax Allocation Rules

SAP provides subschema QPCD (with rules QP10, QP12, QP14, QP16, QP18, and QP20) to handle the employer (ER) benefits and tax distributions.

We have briefly seen the differences in public sector payroll in this section. In the next section, we will similarly review the non-profit organization (NPO) payroll.

4.5 Overview of Non-Profit Organization (NPO) Payroll

Schema UN00 is available in SAP ECC6 to work with non-profit organization implementations. The highlights of this payroll are as follows. Please note that there are new infotypes in series 900 that are provided for NPO payroll in SAP. The subschemas listed below discuss a brief overview of specific NPO functionality. Similar to the normal U.S. schema, much of the functionality such as retroactive accounting, factoring, proration, and so on are available. In the IMG you will find a Payroll-Non-Profit node with all of the configuration underneath. Our focus in this book is US Payroll, but this section has been added to make you aware of this functionality. Similar to our learning from earlier chapters, you can continue to explore this functionality if you need to analyze it in detail. The new infotypes can be configured by following the IMG path Payroll Non-Profit Organization • Payroll Master Data.

The following infotypes are relevant for NPO payroll in the SAP system:

- Subschema UN03 manages various allowances such as dependent, hardship, higher duty, mobility, etc. This subschema is delivered with new rules to manage these allowances.
- Subschema UNLN processes Infotype 0045 (advance payments and recovery).
- Subschema UN21 processes not only normal payments and deductions (Infotypes 0014/0015), but also Infotype 0965, education grant.
- Subschema UN23 processes rental subsidies using Infotype 0962 and has functionality for processing broker fees related to rentals.
- Subschema UN24 processes pensions and pension funds using Infotype 0961. Both employee and employer portions can be handled.

Now that we have discussed schemas, it is time for us to move to rules. Schemas and subschemas contain rules, and the rules can be both SAP delivered as well as custom created by configurators. In the next section, we will discuss plenty of examples to learn the payroll rules.

4.6 Writing Rules in US Payroll

In Figure 4.1, you saw that rules are related through functions in the schema, and they use operations to execute the logic. Rules provide tremendous flexibility to SAP payroll configurators for managing business requirements. In fact, rules make it very easy to fulfill payroll requirements, which typically depend on industry-specific issues or union agreements. SAP defines rules (also known as *personnel calculation rules*, or PCRs) as statements intended "for the execution of defined tasks in time management and payroll."

While the discussion here is limited to payroll rules, the philosophy applies to both time management and payroll rules. Rules manipulate wage types using operations. Appendix F lists many useful operations for rules. Although we will be learning about many operations in the examples in this section, it is not possible to cover all of the operations SAP delivers. So use the examples in Appendix F and practice writing rules using other operations.

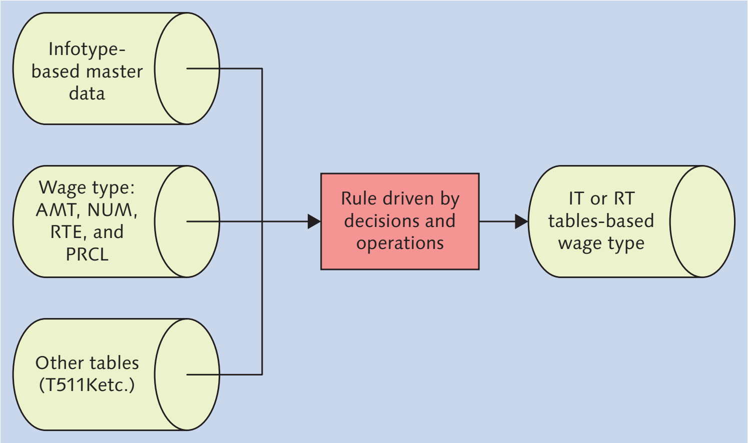

Infotype-based master data
Wage type: AMT, NUM, RTE, and PRCL
Other tables (T511Ketc.)
Rule driven by decisions and operations
IT or RT tables-based wage type

Figure 4.11 Rules Processing—Input and Output

As you read the examples in this section, you will notice that rules are based purely on logic and arithmetic. Once you have figured these two out, you will be able to code error-free rules. Figure 4.11 shows the simple concepts of processing the rules, which involves input, processing, and output. The three basic elements of wage types seen in Chapter 3 are RTE (rate), NUM (number), and AMT (amount); they are used heavily in many rules. Table T511K in Figure 4.11 maintains various constants, and you can also create your own constants in this table and use them in rules.

Let's now move on to rules editing, where you'll learn how to create and maintain rules.

4.6.1 The Rules Editor

Transaction PE02 gives you access to the Rules Editor. When you start this transaction, you will notice that each rule has three important dimensions:

- **Attributes**
  Name and creation date for the rule
- **Source text**
  Actual logic, with operations
- **Documentation**
  Documents the functionality

These three dimensions contain all of the information about a particular rule. As mentioned earlier, do not modify SAP-delivered rules; instead, always follow the *copy and modify* approach. You can also access rules from the Schema Editor (Transaction PE01). For example, explode the subschema and double-click on the rule to open Transaction PE02, the Rules Editor. The rule source text is always valid for a combination of wage types and employee subgroup groupings. Before proceeding to create a rule, it's important to understand the combination of employee subgroup and wage types.

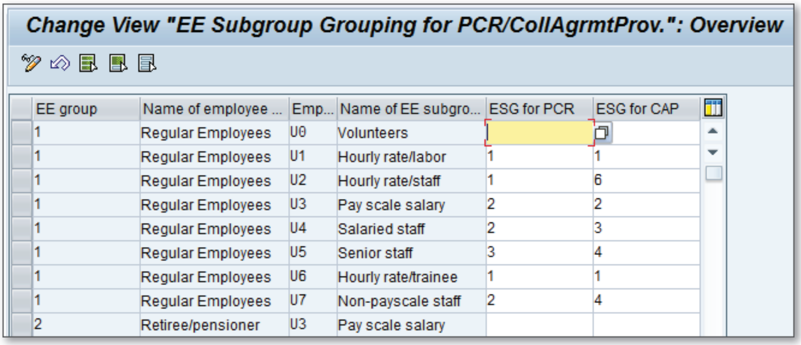
> **Note**
>
> You can use the Rules Editor with one of two display options: the table display or the structure graphics display. The examples that follow use the table display option.

Wage Types

You are already familiar with wage types from Chapter 3. You can code rules for any valid wage type (remember that the logic in the rule will only work when payroll processing finds that wage type). You can use the wild-card characters **** in this field to specify that the rule will work for all wage types. For example, if you insert a rule into the schema, and the input table to the rule reads the wage type, then the rule logic checks to see if the wage type is valid for processing. If you have used ****, then all wage types are processed.

Employee Subgroup Grouping (ESG)

Let's detour in the IMG menu to Personnel Management • Personnel Administration • Basic Pay • ESG for the PCR node and refer to Figure 4.12 for the employee subgroup grouping for PCRs (payroll calculation rules). Notice that the HR enterprise structures (employee groups, employee subgroups) have an impact on the grouping. For example, hourly employees have a different grouping than salaried employees. Therefore, if you need to write certain rules in payroll and differentiate between certain employee types, you can either do it through these groupings or by having separate wage types. Another example might be where delivery drivers at a pizza company receive a special allowance, and the calculation of this allowance depends on the group of employees called drivers. In this case, if you want the rule to apply to all types of employees instead of just the drivers, then you can use the wildcard * in the ESG field in the Rules Editor.



Change View "EE Subgroup Grouping for PCR/CollAgrmtProv.": Overview

EE group	Name of employee ...	Emp...	Name of EE subgro...	ESG for PCR	ESG for CAP
1	Regular Employees	U0	Volunteers		
1	Regular Employees	U1	Hourly rate/labor	1	1
1	Regular Employees	U2	Hourly rate/staff	1	6
1	Regular Employees	U3	Pay scale salary	2	2
1	Regular Employees	U4	Salaried staff	2	3
1	Regular Employees	U5	Senior staff	3	4
1	Regular Employees	U6	Hourly rate/trainee	1	1
1	Regular Employees	U7	Non-payscale staff	2	4
2	Retiree/pensioner	U3	Pay scale salary		

Figure 4.12 Configuration for Employee Subgroup Grouping

In US Payroll, you will find many situations dealing with unions and different employee groupings that require using this field. As such, it is a good idea to check the payroll processing requirements while finalizing your enterprise structures. The best way to learn about rules is to create one, as discussed in the next section.

4.6.2 Creating Simple Rules

This section follows a step-by-step approach to creating a simple rule with basic operations:

1. **Use Transaction PE02 to access the Rules Editor.**
   Enter the name of the rule and click on the CREATE icon; the Editor takes you to the attributes screen, as shown in Figure 4.13.

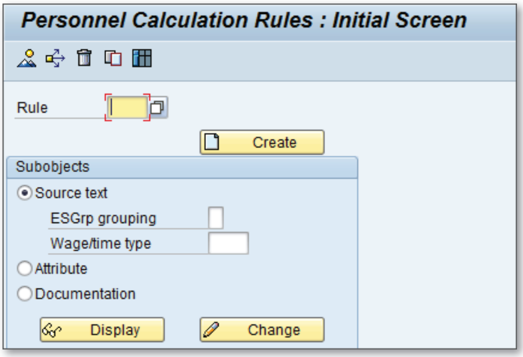


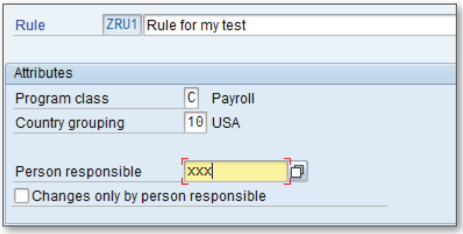
Figure 4.13 The Rules Editor

2. **Create the rule's attributes.**

   Create these attribute fields, as shown in Figure 4.14:

   - PROGRAM CLASS = C for payroll rules
   - COUNTRY GROUPING = 10 because you are working with US Payroll
   - **Text**
     Provide a useful description for the rule in the field next to the name of the rule, as shown in Figure 4.14. Sometimes, four-character naming is not enough because most of the common rules start with Z, and three characters may not be enough to assign a descriptive name to your rule.
   - **Flag**
     Define who can make changes to your rule. If possible, do not check this, so other configurators can modify your rule, if required. In this example, we are creating a rule called ZRU1, which is used in this payroll run. The name ZRU1

has no significance. You can use your own innovative naming scheme so the four-character name meets your needs.



Rule ZRU1 Rule for my test

Attributes

Program class C Payroll

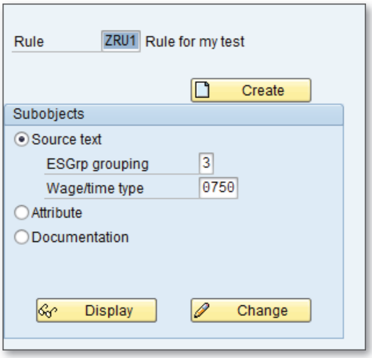
Country grouping 10 USA

Person responsible xxx

☐ Changes only by person responsible

Figure 4.14 Rule Attributes

3. **Access the editor for a combination of ESG and wage type.**
   Section 4.5.1 discussed ESG and wage types—the two key fields used to write your source code for the rule. Create this rule for ESG = 3 and WAGE TYPE = 0750. This means the rule only works when an employee in payroll processing falls in ESG 3 and has wage type 0750 coming in for processing. Later in the chapter you will learn how the rule reads wage types for input processing. As shown in Figure 4.15, you can start the rule editor by filling in ESG and wage type fields.



Rule ZRU1 Rule for my test

Subobjects

● Source text

ESGrp grouping 3

Wage/time type 0750

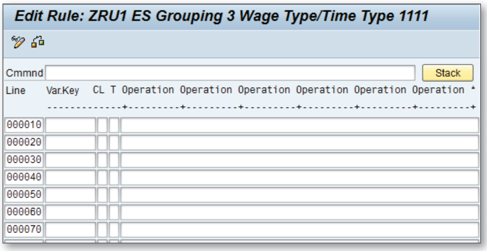
○ Attribute

○ Documentation

Figure 4.15 Creating a Rule for ESG–Wage Type Combination

   When you enter the ESG (3) and wage type (in this example, wage type 1111), the blank editor screen appears, as shown in Figure 4.16.

4. **Create source code statements in the rule.**
   Figure 4.16 shows the blank editor screen in which you create the statements. Note the heading for the rule editor at this stage; it shows that you are writing code for a particular combination of ESG and wage types.



Edit Rule: ZRU1 ES Grouping 3 Wage Type/Time Type 1111

```
Line    Var.Key  CL T Operation Operation Operation Operation Operation Operation *
        ------------+---------+---------+---------+---------+---------+---------+
000010
000020
000030
000040
000050
000060
000070
```

Figure 4.16 Editor Screen for ESG = 3 and Wage Type = 1111

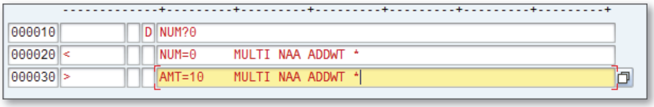
Next, start coding the lines in the editor screen. Remember, the only way to learn different styles and logics of the payroll rules is to try different examples. You might feel somewhat lost at the beginning, but don't worry. You will get a feel for the process when you complete the coding and run the rule. Before you start coding (as you already know from the discussion of fields in the Schema Editor), you need to become familiar with the Rules Editor fields as explained here:

- **VARKEY (variable key)**
  Sends the logic of the rule to different routes based on the decisions you make in the rule
- **NL (next line)**
  If your logic exceeds one line, you can go to the next line (examples forthcoming)
- **T**
  Type of rule
- **D**
  Decision on the variable key
- **P**
  Rule for which processing continues to the next operation on the line
- **Z**
  Rule for which processing does not continue to the next operation and logic jumps. Later, you will see an example to clarify the type P and Z
- *****
  Comment line
- **OPERATION**
  Starts at the correct indentation, where you see a plus (+) sign on the top line

- **Line numbers**
  Guide line sequencing for the code (as was the case with the Schema Editor)

5. **Plan source code of the rule.**
   Figure 4.17 shows that a few lines have been added to the source code. In the first line, an instruction is made in the NUM field to check if the number is nonzero. In subsequent lines, depending on the value of NUM, certain operations will be performed. If the number is greater than zero, then the line 000030 logic will work; if the number is less than zero, then rule line 000020 will work. Please refer to Appendix F for a list and description of various operations you can use in rules. Operation MULTI is used to multiply NUM (N) with AMT (A), which moves the result into the AMT (A) field.

At the end of the code, click on the balancing icon to see if there are any syntax errors; if not, save the rule. Please note the Rule OK message on the last line. The message shows that there are no syntax errors and the rule can be used in the schema.



```
            -------------+---------+---------+---------+---------+---------+---------+
000010          D NUM?0
000020 <          NUM=0     MULTI NAA ADDWT *
000030 >          AMT=10    MULTI NAA ADDWT *
```

Figure 4.17 Adding Source Code Lines to the Rule

Note

Remember the operations of a rule and always use ADDWT * logic in the rules; otherwise, you will run a risk of losing wage types during the rule processing. Wage types that the rule does not explicitly refer to are not written to the IT table if you do not have the ADDWT * logic.

4.6.3 Creating Complex Rules

It's now time to move on to coding more complex rules. In the interest of space, let's keep the source code and logic part unchanged, to avoid explaining the routine elements of rules we have already seen. Instead, we will discuss different scenarios with rules. By no means is this a complete list, but it will definitely help you build confidence as you create new rules of your own. For a list of additional available operations, see Appendix F.

The payroll rules and some of the complex operations contained in them are required in many day-to-day situations. These situations broadly fall into the following categories:

- Calendar-based manipulations in payroll (for example, with certain payroll periods during the year)
- Employee type-based manipulations (for example, with certain employees having special calculations)
- Creating new wage types in payroll based on calculations (for example, take 10% of gross earnings and create a wage type as the basis for other subsequent calculations)

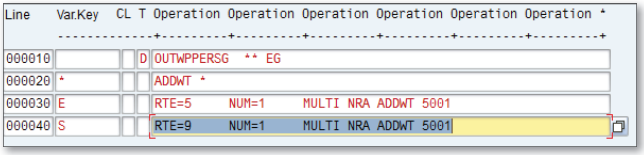
The more rules you try and test, more you will learn about rules. Every SAP Payroll implementation can have its own unique set of custom rules, which can be different from other implementations. In the next section, we will use different examples for coding these rules. These examples will serve as guidance and a starting point in your journey to handle rules.

Example 1: Enterprise Structure-Driven Requirement

The enterprise structure-driven requirements are typically based on location, employee groupings, or union differentiators (if unions are separated in enterprise structures). Here are some examples that drive these rules:

- Salaried and hourly employee separation for calculations
- Different unions' different rates for calculations
- Different locations' varying treatment
- Separate treatment of employees (executives and non-executives)

Figure 4.18 shows one such example, using the operation OUTWPPERSG, which reads the employee group and separates the logic for calculation. The operation OUTWP has around 30 variations to read different elements of enterprise structures, payroll structures, and organization structures to make the decisions in the rule.



Line	Var.Key	CL	T	Operation Operation Operation Operation Operation Operation *
000010			D	OUTWPPERSG ** EG
000020	*			ADDWT *
000030	E			RTE=5 NUM=1 MULTI NRA ADDWT 5001
000040	S			RTE=9 NUM=1 MULTI NRA ADDWT 5001

Figure 4.18 Enterprise Structure Decision

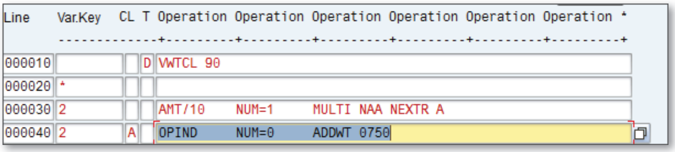
Table 4.7 presents a line-by-line analysis of the rule shown in Figure 4.18. Employee groups E and S are used only as examples and can have any meaning from project to project.

Line Sequence Number	Operation/Logic Explanation
000010	Decision on employee group. Note that you can write a comment if you skip the operation indentation, as shown by the ** EG comment in this line.
000020	If the employee group is * (which means any value), then do not do anything. Operation ADDWT * writes back the wage type.
000030	If the employee group is E, then set the rate = 5 and number = 1. Multiply number with rate and move the result to the Amount field. Specify wage type 5001, which means wage type 5001 will have RTE = 5, NUM = 1, and AMT = 5.
000040	Same as for 0000030, except that when employee group = S, RTE = 9.

Table 4.7 Rule Logic Using the OUTWP Operation

Example 2: Wage Type-Driven Requirement

Many requirements are driven by a certain wage type and its processing classes. As discussed earlier in the chapter, processing classes can be used to determine many different dimensions of a wage type. As such, SAP-delivered rules, customized rules, and processing classes form popular decision-making criteria. Figure 4.19 shows a decision based on processing class 90. The operation VWTCL reads the processing class of any wage type that may be getting processed in this rule. Similarly, you can write a rule to make a decision on other processing classes using the operation VWTCL. It is also possible to create new specifications for any processing class and then use that value in a rule.



```
Line    Var.Key   CL T Operation Operation Operation Operation Operation Operation *
        ------------+---------+---------+---------+---------+---------+---------+
000010              D VWTCL 90
000020  *
000030  2              AMT/10    NUM=1     MULTI NAA NEXTR A
000040  2          A   OPIND     NUM=0     ADDWT 0750
```

Figure 4.19 Processing Class Decision

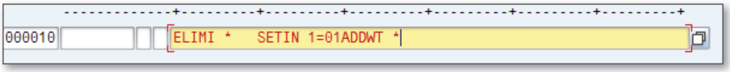
Table 4.8 presents the line-by-line analysis of the rule shown in Figure 4.19. Depending on the specification value of a processing class, the rule logic routes to different lines. Please remember that an asterisk (*) is used as a wild-card character in all areas of rule writing.

Line Sequence Number	Operation/Logic Explanation
000010	Decision based on processing class 90. Note that the rule only works for wage type = 0750 and employee subgroup grouping = 3 (which is typically reserved for salaried employees).
000020	If the employee group is * (which means any value), then don't do anything. This should have been the ADDWT * operation. If you leave this value blank, you will discover that the wage type gets "dropped" when the rule is processed; if the condition is met, the wage types need to be written back into the table.
000030	If the specification for processing class 90 = 2, divide the AMT field by 10, multiply NUM with AMT, and move the result into AMT.
000040	Because of the NEXTR operation from an earlier line, the logic continues in this line. The OPIND operation changes the sign for the amount. NUM = 0 sets zero in the NUM field and writes wage type 0750 with the ADDWT operation. As such, wage type 0750 will have NUM = 0, and AMT will be divided by a factor of 10 from the original amount in wage type 0750.

Table 4.8 Rule Logic Using the VWTCL Operation

Example 3: Adjusting Splits of Wage Types

As shown earlier for function WPBP, there can be splits to the wage types if there are changes to Infotypes 0001, 0027, 0007, and 0008 in the middle of a pay period. When the wage type has split, you will see two, three, or as many occurrences as the number of splits for the wage types. Wage type splits have an impact on reports, such as wage type journals, and they can have an impact on financial postings. As a result, SAP provides some popular operations to manage splits. Figure 4.20 shows the operations ELIMI and SETIN; together they will help you eliminate and reset appropriate splits. A discussion of the types of splits is beyond the scope of this book. If you need to learn more about them, refer to SAP's documentation for the ELIMI operation. There are many different types of splits, such as A for work center, K for cost center, and so on. We recommend that you use Transaction PE04 and access the documentation for operation ELIMI to read more about splits.

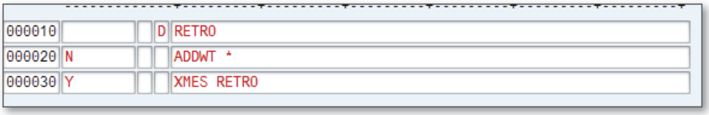


```
000010      ELIMI *   SETIN 1=01ADDWT *
```

Figure 4.20 Eliminating Wage Type Splits

Example 4: Checking if Payroll Run Has Retro

Retroactive accounting during payroll processing is a very common phenomenon. Figure 4.21 presents an example to see whether retroactive accounting is taking place in the payroll process and then drives the logic based on the result. Operation RETRO uses yes or no decisions to drive the logic, as shown in Figure 4.21.

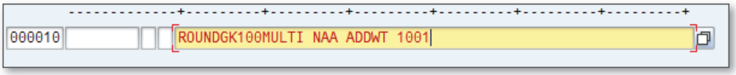


```
000010     D RETRO
000020 N     ADDWT *
000030 Y     XMES RETRO
```

Figure 4.21 Checking a Retro Run

Example 5: Managing Rounding of Values

In payroll, you will likely run into situations where you need to round items, such as salary amounts, hours, or other numbers associated with wage types. ROUND is a very simple operation that handles rounding. Figure 4.22 uses the ROUND operation to round off the numbers and then writes the wage type after the calculation. You can read more about the ROUND operation using Transaction PE04.

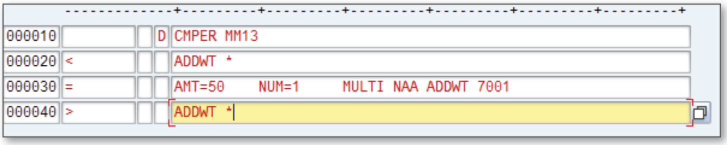


```
000010      ROUNDGK100MULTI NAA ADDWT 1001
```

Figure 4.22 Rounding Wage Types

Example 6: Using Payroll Periods

Some industries have unique requirements that, in a particular pay period, are used to manage certain earnings or deductions. Some examples include performance bonuses and executive compensation such as profit sharing. As far as deductions are concerned, you learned about payment models in Chapter 3. Payment models manage the deductions by a predefined pay period-based calendar. However, payment models also have their own maintenance based on calendar years. You can check payroll periods using a lesser-known operation called CMPER. Figure 4.23 shows the rule that checks for payroll period 13. If the current period is 13, the rule gives $50 through wage type 7001 to all employees. Table 4.9 lists the line-by-line analysis of this rule.



```
000010      D CMPER MM13
000020 <      ADDWT *
000030 =      AMT=50    NUM=1     MULTI NAA ADDWT 7001
000040 >      ADDWT *
```

Figure 4.23 Using the Payroll Period for a Decision

Line Sequence Number	Operation/Logic Explanation
000010	Decision on payroll period value. The operation CMPER can be used in more than one way to read the current period or a retro period. This line is reading the current period.
000020	If the value of the payroll period is less than 13 (such as if payroll is running for period 11), then the rule won't do anything and will just write back the wage types by using the ADDWT operation.
000030	If the period *is* 13, then the rule creates wage type 7001 with AMT = $50, which is created with the MULTI operation.
000040	If the period is greater than 13, then just like line 20, the rule won't do anything.

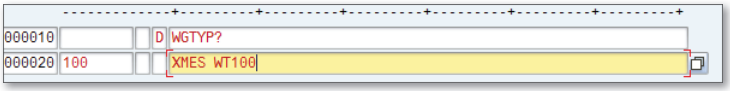
Table 4.9 Rule Logic to Compare the Payroll Period

Example 7: Using Rules to Identify Wage Types or Occurrences of Conditions

The XMES operation helps you print small messages in the payroll log, depending on the decision the rule makes. You can use this operation in many situations such as the following:

- You want to find out if an employee is processed with a certain local tax authority, which is normally not a routine item in master data.
- You want to know when a certain wage type has a negative value; it may be nearly impossible to identify this situation in a large employee population.

In Figure 4.24, if the rule finds wage type 0100 in payroll processing, the payroll log prints the message WT100 in the log.



```
-------------+---------+---------+---------+---------+---------+---------+
000010          D WGTYP?
000020 100        XMES WT100
```

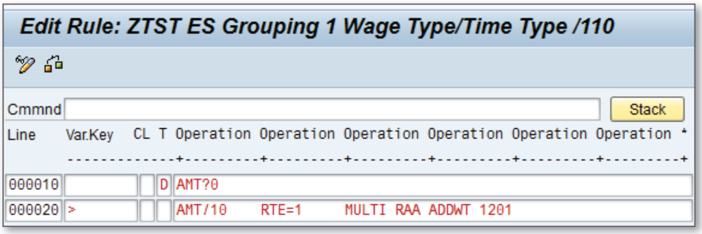
Figure 4.24 Message in Payroll Log when Wage Type is Found

Example 8: Calling One Rule from Another

You can also send the logic from one rule to another. You can either send the logic so that the control is passed to the other rule or you can bring the control back to the same rule after executing the second rule. In the decision logic, if you use rule type P, then the processing logic will come back to the next line in the rule. If you use rule type Z, then the processing will jump to the other rule.

Example 9: Using a Base Technical Wage Type to Derive a New Wage Type

You can use an existing or newly created technical wage type to create and calculate a new wage type. In the example shown in Figure 4.25, technical wage type /110 is used only for a certain type of employees. This example uses /110 (total deduction). However, you can create any new cumulation in your system to create a new technical base wage type. Table 4.10 describes the lines from your rule.



Edit Rule: ZTST ES Grouping 1 Wage Type/Time Type /110

Cmmnd

```
Line    Var.Key   CL T Operation Operation Operation Operation Operation Operation *
             -------------+---------+---------+---------+---------+---------+---------+
000010              D AMT?0
000020 >              AMT/10    RTE=1     MULTI RAA ADDWT 1201
```

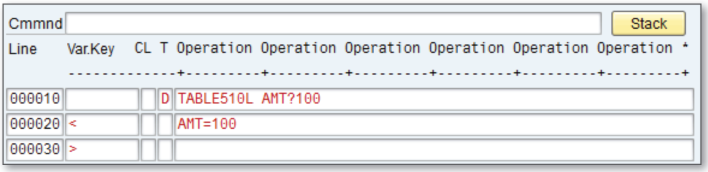
Figure 4.25 Using a Technical Cumulation Wage Type for Creating a New Wage Type

Line Sequence Number	Operation/Logic Explanation
000010	Decision based on non-zero amount. Please note that the rule is coded only for ESG = 1 and wage type = /110 as shown on the top heading.
000020	Amount is divided by 10. Rate is set to 1.0. New wage type 1201 is created in an IT table.

Table 4.10 Creating a Wage Type from a Technical Wage Type

Example 10: Manipulating Infotype 0008

The example in Figure 4.26 shows the use of an operation TABLE, which lets you read infotypes within the payroll rule. This operation can be used with only certain tables in the SAP Payroll system. You can also refer to the documentation of this operation by using Transaction PE04 to list the allowed tables.

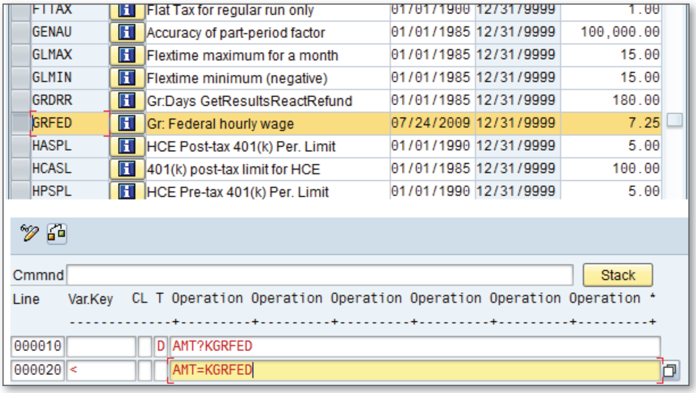


```
Cmmnd                                                          Stack
Line     Var.Key   CL T Operation Operation Operation Operation Operation Operation *
         -------------+---------+---------+---------+---------+---------+---------+
000010               D TABLE510L AMT?100
000020 <                AMT=100
000030 >
```

Figure 4.26 Use of an Operation Table

Example 11: Checking Minimum Wage Rate from Table T511K

In Figure 4.27, table T511K, a minimum wage rate of $7.25 is displayed in the upper half of the screen. The lower half shows a use of this constant (KGRFED) in a rule. The example shows a comparison of the rate; if the rate is less than the minimum wage rate, then it is set to the minimum by using the T511K constant from the constants table.



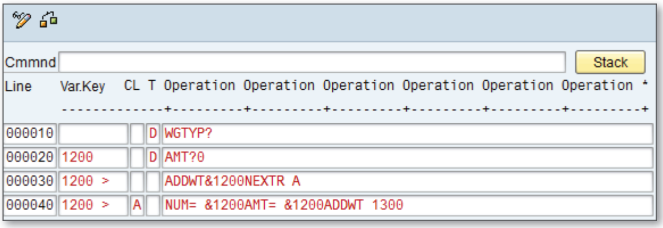
FTTAX	i	Flat Tax for regular run only	01/01/1900	12/31/9999	1.00
GENAU	i	Accuracy of part-period factor	01/01/1985	12/31/9999	100,000.00
GLMAX	i	Flextime maximum for a month	01/01/1985	12/31/9999	15.00
GLMIN	i	Flextime minimum (negative)	01/01/1985	12/31/9999	15.00
GRDRR	i	Gr.Days GetResultsReactRefund	01/01/1985	12/31/9999	180.00
GRFED	i	Gr. Federal hourly wage	07/24/2009	12/31/9999	7.25
HASPL	i	HCE Post-tax 401(k) Per. Limit	01/01/1990	12/31/9999	5.00
HCASL	i	401(k) post-tax limit for HCE	01/01/1985	12/31/9999	100.00
HPSPL	i	HCE Pre-tax 401(k) Per. Limit	01/01/1990	12/31/9999	5.00

```
Cmmnd                                                        Stack
Line     Var.Key   CL T Operation Operation Operation Operation Operation Operation *
         -------------+---------+---------+---------+---------+---------+---------+
000010               D AMT?KGRFED
000020 <               AMT=KGRFED
```

Figure 4.27 Using the T511K Constants Table in a Rule

Example 12: Using Temporary Variables in a Rule

Figure 4.28 shows that you can use one wage type to create a temporary variable and then use the details from the temporary variable to create a new wage type. Table 4.11 lists the lines from the rule and the explanation.



Line	Var.Key	CL	T	Operation Operation Operation Operation Operation Operation
000010			D	WGTYP?
000020	1200		D	AMT?0
000030	1200 >			ADDWT&1200NEXTR A
000040	1200 >	A		NUM= &1200AMT= &1200ADDWT 1300

Figure 4.28 Using "&" to Create a Temporary Variable Wage Type

Line Sequence Number	Operation/Logic Explanation
000010	Decision on wage type
000020	If the wage type = 1200, then check the amount field
000030	If the wage type = 1200 and if the amount in the wage type is greater than zero, then create temporary variable &1200. The NEXTR operation passes the rule control to the next line that is identified with "A" as per the NEXTR A operation
000040	Use the NUM and AMT from temporary variable &1200 and create and write a new wage type 1300 in the IT table

Table 4.11 Using the Temporary Wage Type Variable

Now that you have seen a few examples of rule writing, we'll show you how to run them in a schema and perform a drill-down to check results. The process is the same as shown earlier for schema runs.

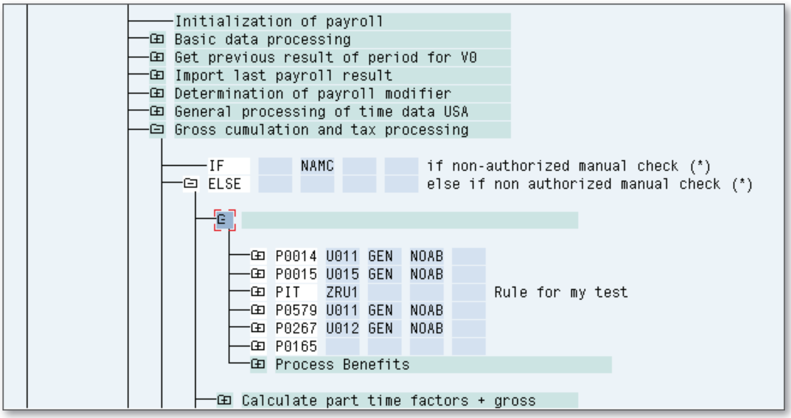
4.6.4 Running Error-Free Rules

After coding the rules you want to use, you need to add them to the correct subschema and test the schema for the desired effect. The obvious question is *How do I decide where to add the rule in the schema?* The answer is straightforward and depends on the answers to the following questions:

- What wage type am I processing in this rule?
- Has the wage type entered through an employee infotype? If yes, have we processed the infotype yet?
- Does this have an impact on taxation? If it does, do I need to process it before tax processing?
- Does this have an impact on retroactive accounting? If so, how do I position it?
- Is this related to an earning, deduction, or tax wage type?

For you to get a better understanding of how to run the rules, let's walk through the process step by step:

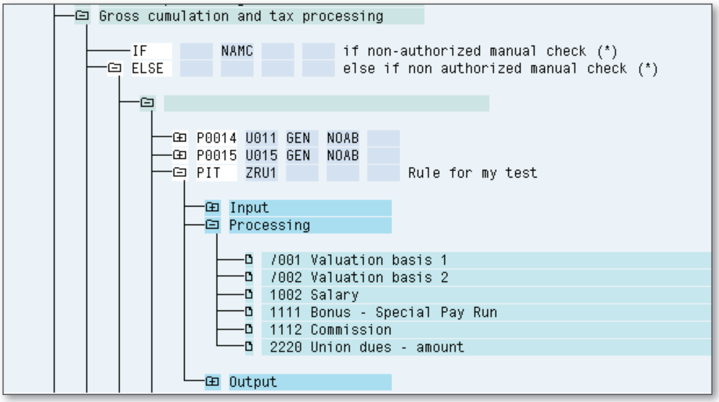
1. **Add rules to the subschema.**
   Add the PIT function to the schema's rule. Figure 4.29 shows how the PIT function is used to add rule ZRU1 to the schema.



```
Initialization of payroll
Basic data processing
Get previous result of period for V0
Import last payroll result
Determination of payroll modifier
General processing of time data USA
Gross cumulation and tax processing

  IF     NAMC       if non-authorized manual check (*)
  ELSE              else if non authorized manual check (*)

      P0014 U011 GEN  NOAB
      P0015 U015 GEN  NOAB
      PIT   ZRU1             Rule for my test
      P0579 U011 GEN  NOAB
      P0267 U012 GEN  NOAB
      P0165
      Process Benefits

  Calculate part time factors + gross
```

Figure 4.29 Adding a Custom Rule to the Schema

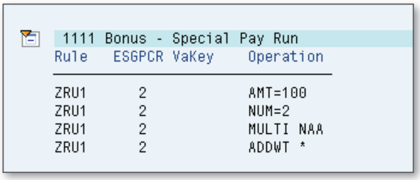
2. **Check if the input IT table is sending the correct wage type to the rule.**
   Figure 4.30 shows the drill-down when the schema is run. As covered earlier in the discussion about writing rules, you need to focus on the input, processing, and output elements.



```
Gross cumulation and tax processing
  IF        NAMC        if non-authorized manual check (*)
  ELSE                  else if non authorized manual check (*)

      P0014 U011 GEN  NOAB
      P0015 U015 GEN  NOAB
      PIT   ZRU1                Rule for my test
        Input
        Processing
          /001 Valuation basis 1
          /002 Valuation basis 2
          1002 Salary
          1111 Bonus - Special Pay Run
          1112 Commission
          2220 Union dues - amount
        Output
```

Figure 4.30 Drill-Down Custom Rule in Runtime

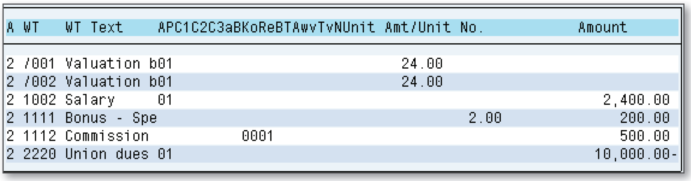
3. **Drill down into the processing.**
   When you drill down at the processing of the rule (shown in Figure 4.27), the line sequencing-based logic is visible. In the example in Figure 4.31, an employee that falls under ESG 2 and has wage type 1111 is processed. As such, you can check to see if the rule logic is actually working while the rule is being executed in the schema.



```
1111 Bonus - Special Pay Run
Rule   ESGPCR VaKey    Operation

ZRU1      2            AMT=100
ZRU1      2            NUM=2
ZRU1      2            MULTI NAA
ZRU1      2            ADDWT *
```

Figure 4.31 Inside the Rule During Runtime

4. **Verify whether the ouput IT table has the correct values.**
   This step shows you the wage types with rates, number, and amount as processed by the rule. Figure 4.32 shows the IT table from a rule. IT tables have wage types with RTE, NUM, and AMT values.



```
A WT   WT Text     APC1C2C3aBKoReBTAwvTvNUnit Amt/Unit No.             Amount

2 /001 Valuation b01                             24.00
2 /002 Valuation b01                             24.00
2 1002 Salary     01                                                 2,400.00
2 1111 Bonus - Spe                                        2.00         200.00
2 1112 Commission          0001                                        500.00
2 2220 Union dues 01                                               10,000.00-
```

Figure 4.32 Input Table (IT)

During our journey through SAP payroll, we've always heard that SAP payroll is all about schemas and rules. Although this chapter is long and detailed, we hope it provides the necessary details that you seek. The examples in this section provide good guidance on how to code the rules and how to use the operations.

4.7 Summary

You have learned a lot about schemas, subschemas, rules, and operations in this chapter. You can now try to build your own rules as you further explore this topic. The sections on the runtime environment will help you debug your rules and make changes to schemas.

In this chapter, we talked about benefits processing in schemas. In the next chapter, you will learn more about how to integrate benefits with your payroll. The earlier discussion of wage types, along with the current discussion of schemas, and the following discussion of benefits will make the discussion of benefits processing almost complete. Subsequent chapters related to garnishments (Chapter 7) and tax processing (Chapter 8) will continue to reference schemas and discuss rules.

This brief overview of U.S. public sector functionality as well as non-profit organization functionality will be useful to you if your implementation leverages any of those areas due to the nature of the business. The SAP core structure of payroll drivers, schemas, rules, benefits integration, and time integration remains the same, irrespective of the country or type of payroll handled in the SAP system. Therefore, you should be able to apply that learning and discussion if you need to explore these new territories.

Index

401(k), 26, 62, 84, 88, 183
- *Benefits, 25*
- *Catch-up contributions, 26, 178, 183, 188*
- *Change in contribution, 48, 50, 198*
- *Contributions deduction, 67, 68*
- *Deduction changes, 356*
- *Deductions, 68*
- *Deductions infotype, 245*
- *Deductions wage type, 250*
- *Wage type, 80*
- *Wage type assignment, 189*

1042S
- *Processing, 246*
- *Statutory reporting, 153*

1099
- *Overpayment adjustments, 279*

A

Absence, 201
Accounts Payable (AP), 26, 177, 224, 317
ACCR table, 302, 303
Accruals, 26, 298, 303
- *Configuration, 299*
- *Defined, 298*
- *Process and posting, 302*
- *Subschema, 299*

Acknowledgment program, 318
ADDCU, 136, 439
Additional master data infotypes, 38, 39
ADDWT, 439, 441
ADDWTE, 440
Ad hoc query
- *Reporting, 58*

ADJT subtype, 260, 261
Adjustment, 267
- *Adjustment date, 262*

ADP, 28
Advanced topics, 279
After-tax, 432
After-tax deduction
- *Wage type, 193*

Allowable disposable net income, 228
Amount, 70
AMT, 440
- *Wage type element, 70*

Annual report, 268
Arrears
- *ARRRS table, 107, 138, 141*
- *Net pay, 93*
- *Processing, 86, 104, 105, 106, 123*
- *Retroactive, 106*

Attendance, 201
Audit report, 268
Author's email address, 360

B

B2 time clusters, 136
Balance, 432
- *Goal amount, 86*

Balance and deduction, 86
Balance sheet, 112
- *Account, 68*

Bank account
- *Direct deposit, 41*

Bank transfer, 301, 373
- *Net processing, 142*

Basic pay, 40, 66, 150
- *Infotype, 245*
- *Wage type for, 89*

Benefit, 133, 139
- *Enrollment data, 322*
- *Health insurance, 25*
- *Life insurance, 25*
- *Pension, 25*
- *Provider, 197*

Benefits, 25
- *Data, 36*
- *Deductions, 409*
- *Different cost types, 180*
- *Flexible spending accounts, 25*

Benefits constants table
- *Shortcut, 186*

Benefits-payroll integration, 175
Benefits-related wage types, 409
Best practices, 358
Schemas, rules, and wage types, 358
Bill of materials
Schema, 128
Biweekly payroll, 103, 126, 298
Bond purchase, 411
Bonus
Pay, 71
Payment, 51, 260
Payroll run, 323
Bonus and off-cycle, 51
BSAL
Base salary wage type, 190
BSI software, 271
BSI Tax Factory, 27, 28, 31, 80, 89, 92, 138, 241
BT table, 373
Business Software Inc., 27

C

C1 table, 216
Cafeteria plans, 434
Call-in time, 366
Cash advance processing, 88
Cash-based accounting, 298
CATS, 203
Ceridian, 28
Check date, 301
CHECK function, 134
Check processing, 373
Child support, 225
Multiple payments, 237
Claim clearing deduction, 419
Claim generation process, 280
Claims, 266, 280
Handling, 279, 325
Identifying in payroll, 282, 341
Strategy, 284
Claims processing
Diagram, 280
Claims report, 280, 282, 283, 284, 287
Clearing account, 122
Clearing claims, 285, 286
Employee payment plan, 288
Payroll-forgiven, 284
Payroll-not-forgiven, 285
Closing date, 301
Clusters, 31
Time, 136
Cluster table CLM, 294
CMPER, 440
Compare payroll periods, 440
Comments
Asterisk, 363
Common paymaster, 274
Constants, 192
Defined, 186
Table, 186
Contribution limits
HCA and DCA, 195
Controlling (CO), 26
Control record, 29, 357
Bypassing the check, 134
Payroll calendar, 27
COPY function, 130
Cost Center Accounting (CO-CCA), 26
Costs tab
Infotype 0167, 180
Country code 10, 71
Coverage amount, 191
Cross-year tax calculation, 36
CRT, 102, 177, 439
Payroll table, 259
Results table, 312
Cumulation, 62, 67, 69, 79, 80, 81, 82, 83, 84, 88, 101, 102, 140, 368, 430
Year-to-date, 31
Cumulative results table, 102, 143
Current calendar
Get month or week, 447
Customer name space
Constants, 187
Custom naming specifications, 131
Custom schemas
Defined, 126
Custom tax models
Creating, 250
Custom wage types, 409

D

Data errors in payroll
  Table, 56
Data Medium Exchange (DME), 94
  Workbench, 94
Decision logic, 447
Deduction model
  Annual maintenance, 103
Deductions, 83
  Calculation, 369
  DDNTK, 106, 141, 370, 372
  Defined, 66, 83
  Employee-dependent, 37
  Involuntary, 33
  Minus sign, 83
  Not taken, 104, 106
  Post-tax, 63
  Pre-tax, 63
  Priority A, 105
  Priority and arrears, 104
  Priority levels, 105
  Remittance to vendors, 312
  Taxes, 62
  Technical wage types, 88
  Voluntary, 33
Deductions and earnings
  Infotype, 49
Deduction wage type, 86, 366
  Configuration steps, 85
Department of Health and Human Services, 229
Dependent care account (DCA), 195
Dialogue, 75
Dialogue wage type, 66
  Creating, 75
  Deductions, 83
Direct deposit, 41, 94
Disposable income, 228
Disposable net income, 26, 82, 142, 226, 228, 236, 371
  Concept, 230
Documentation, 355
Drill-down, 149
  Defined, 135
Due date, 317

E

Earliest retroactive date, 34
Earnings, 40, 75, 307
  Defined, 63, 66
  Entering with infotypes, 77
  Taxes, 63
Earnings and deductions, 368
  Infotype, 49
Earning wage type, 67
  Processing, 82
  M003, 90
EE pre-tax deductions, 282
EIN, 273
Elements
  Wage types, 69
ELIMI, 165, 441, 444
  Eliminate split indicators, 441
Employee cost, 180
Employee data
  Normal, 36
  Transactions, 36
Employee event, 48
  Changes to health plan coverage, 197
Employee master data, 373
  Infotypes, 33, 36
  Infotypes in, 31
Employee remuneration, 42
Employee Self-Service (ESS), 35
Employee tax, 435
Employer contribution
  Wage type, 193
Employer cost, 177, 181
Employer tax, 246, 435
END OF COPY, 366
ERROR, 442
Error conditions, 148
Error-free report
  Test run, 77
Errors
  Detecting, 127
  Reporting, 256
Evaluation classes, 68
Executive employee, 68
Exempt, 202
Expatriate, 432
  Expatriate payroll, 319

Expense, 112
Expense account, 68
Export results table, 374
External bank transfer, 52

F

Factoring, 81
Wage types, 81
Federal Consumer Credit Protection Act (CCPA), 228, 229, 230
Federal Insurance Contribution Act (FICA), 63, 192
Federal taxes
Wage types, 92
Financial Accounting (FI), 26
Financial account management, 314
FI posting, 315
Flexible spending account (FSA), 26, 176, 195, 322
Deduction, 67
Lifecycle, 195
Outsourcing, 196
FLSA, 219, 437
Forgiveness
Write off amounts owed, 287
Form 941, 266
FOR-PERIOD, 440
Functions, 129, 131
CHECK, 134
COM, 363
COPY UBE1, 139
COPY UBE2, 141
COPY UDP0, 141
DAYPR, 136
GEN/8, 140
IF, 142
LIMIT, 141
P0002, 135
P0006, 135
P0009, 143
P9ZNC, 143
P0011, 143
P0014, 135, 139, 151
P0015, 139
P0103, 154
Functions (Cont.)
P0167, 140
P0168, 140
P0169, 141
P0170, 141
P0207, 135
P0221, 137
P0267, 139
P0377, 141
P2003, 136
P2010, 136
PARTT, 136, 365
PRDNT, 141
PRT, 142
UACGF, 302
UGARN, 142
UPAR1, 138
USTAX, 139, 151
WPBP, 134, 165
ZLIT, 136, 365
Funds Management (FM), 26

G

Garnishment, 26, 133, 142, 312, 371, 432
Calculation, 237
Configuring, 224
Data, 36
Document, 224
Impact on earnings and deductions, 232
Infotype 0194, 224
Infotype 0195, 224
Infotype 0216, 224
Limit amounts, 228
Managing wage types, 232
Multiple, 225, 231
Order, 224, 238
Order type, 227
Process flow diagram, 223
Subschema, 142, 233, 238, 371
Garnishment deduction calculation, 238
GCY, 442, 444
General Ledger (GL), 26, 177, 314
Goal amount
Balance, 86
Goal and balance wage type, 86

Goal-related wage type, 366
Goals and deductions wage type, 369
Go-live
Mid-year, 322
New year, 322
Good money, 94, 283
Defined, 94
Wage type, 94
GRORD
Garnishment order table, 238
Gross
Calculation, 133
Taxable, 254
Total, 66
Gross compensation and time
Subschema, 135
Gross earnings and deductions
Wage types, 93
Gross income
Taxes, 371
Gross pay
Earnings, 63
Pre-tax deductions from, 63
Wage type, 80
Gross payroll, 307
Defined, 307
Exporting, 309
Gross to net, 306
Gross up, 320, 410
Group term life insurance, 191
GRREC
Garnishment records table, 238
GRT table, 309, 310

H

Healthcare account (HCA), 195
Health expense
Qualified, 196
Health insurance
Deduction, 67
Health plan, 44, 176, 197
Deductions, 68
Infotype, 245
HIS, 58
HR Payee, 314, 316
Human Capital Management (HCM), 23
Hypothetical tax, 319

I

Identify claims, 289
IMG, 78, 86, 103
Implementation tip, 321
Import menu
Net payroll, 310
Imputed income, 63, 178, 191, 192, 193
Industry Solution-Public Sector (IS-PS), 152
Inflow wage type, 138
Infotype
0003, 34, 285, 356
0006, 43
0008, 33, 40, 63
0009, 41, 373
0011, 52, 373
0014, 44, 49, 63, 87, 103, 104, 288, 367
0015, 50, 63, 87, 288, 367
0041, 178
0050, 214
0057, 44, 63
0167, 44, 176, 179, 367
0168, 191, 367
0169, 50, 63, 176, 184, 369
0170, 176, 196
0172, 196
0207, 43
0208, 43
0209, 243
0210, 53, 55, 243
0221, 286, 366, 379
0234, 243
0235, 244
0267, 51, 284, 285, 287, 289
0377, 367
0909, 294
2010, 39, 42
Infotype 0167
Health plans, 63
Infotype 0168
Insurance plans, 176
Infotype 0170
Flexible spending account plans, 195

Infotype 0194
Garnishment document, 224
Infotype 0195
Garnishment order, 224
Infotype 0207
Residence tax area, 135
Infotype 0216
Garnishment adjustment, 224
Infotype 0221
Creating, 256
Off-cycle payroll, 255
Year-end adjustments, 255
Year-end workbench, 288
Initial tax data, 273
IN-PERIOD, 440
Input combination field, 78
Input table, 150
Insurance plans, 176, 190, 198
Infotype, 245
Integration, 175
Interfaces, 176
Internal Revenue Service (IRS), 46, 63, 67, 244
Investments
Capturing, 185
IT table, 234, 439

K

K operation
RESET, 444

L

Latest document creation dates, 301
Legacy code, 128
Legacy data
Transfer, 322
Loans
Processing personal, 88

M

Mandatory master data infotype, 37, 38
Manual check
Non-authorized, 366, 369
Manual check (Cont.)
Processing, 260
Master data, 36, 307
HR/employee, maintaining, 308
Medicare taxes, 93, 244
Membership fees, 44
Middle of pay period, 368
Model wage types, 66, 71, 76, 77, 83, 181, 189, 193, 232, 284, 288, 409
Catalog, 28, 66
Collection, 66
Commonly used, 409
Copying, 409
Starting with B, 409
Using, 71
Month-end accruals, 298, 306, 373
MULTI
Multiplication operation, 442
Multiplier
RTE, 71

N

Naming
Custom, 131
Negative time tracking, 203, 365
Net calculation, 133
Subschema UNA0, 372
Net pay, 33, 41, 62, 63, 142
Defined, 93
Negative amount, 93
Wage type, 89
Subschema, 142
Net payroll, 306, 307
Defined, 307
Outsourcing, 306
New evaluation run, 317
New overpayments, 291, 419
NEXTR, 442
Next line in a rule operation, 442
Non-authorized check, 137, 287, 366, 369
Non-cash income, 246
Non-exempt, 202
Non-resident alien (NRA), 153, 442
Non-resident taxation, 244, 434
Non-taxable, 246

NUM, 443
Wage type element, 70
Number, 70
Numbering scheme, 72

O

Off-cycle, 51, 126, 262, 264, 323
Off-cycle payroll, 259
Infotype, 255
Infotype 0267, 285
Type A, 285
Type C, 287
One-time deduction, 51
One-time payment
Infotype, 245
Operation, 129, 131, 161
ADDWT, 167
Commonly used, 439
CMPER, 167
DIVID, 441
ELIMI, 165
MULTI, 167
OUTWP, 163
OUTWPPERSG, 163
RETRO, 166
ROUND, 166
SETIN, 165
VWTCL, 164
XMES, 167
Operation limits, 445
OPIND
Operations indicator, 443
Order type, 226
Garnishment, 227
Organizational Management (OM), 15
Output of a payroll process, 31
Outsourcing menu, 307
OUTWP, 443, 446
Workplace and basic pay data, 443
OUTWPPERSG, 443
Overpaid taxes, 283
Overpayment, 255, 256, 279, 325, 435
Claim, 279
Clearing in payroll, 285
Processing, 279
Override methods, 243
Overtime, 102, 366
Pay, 42, 71
Posting to payroll expenses, 68

P

PA0007
Planned work schedule infotype, 446
Part pay period tables, 365
Pay
Basic, 66
Net, 63
Paycheck, 94
Replacement, 260
Pay frequency, 27
Payment
Bonus, 260
Method field, 41
Model, 103
Payment plan, 284
Clearing claims, 288
Payments and deductions
Additional, 50
Pay period, 34
Adjustments, 135
Table, 322
Payroll
Biweekly, 103, 126
Clusters, 31, 374
Cycle, 56
Data, 36
Equation, 33, 62
Indicator, 260
Menu, 282
Off-cycle, 259
Period, 166, 300, 305, 312
Process, 29
Processing, 29, 100, 259, 303
Reconciliation report, 267
Results table (RT), 69, 89, 150, 177, 303, 311, 374, 440
Status, 46, 256, 289
Tables, 259
Weekly, 126
Payroll calculation rules (PCRs), 28, 158

Payroll calendar
Control record, 27
Sample, 45
Payroll control record, 24, 34, 46, 56, 125, 126
Payroll driver, 30, 35, 125, 128, 146, 260, 263
RPCALCU0, 263
Payroll expenses
Posting to a period, 298
Payroll initialization
Subschema, 134
Payroll operations, 439
Payroll results, 87, 90, 100
Reversing, 260
Paystub, 61
Additional features, 64
Example, 61
PCY, 444
PE01, 359
PE02, 359
Pension calculation, 80
Performance reward payment, 51
Permissibility, 66, 68, 79, 98, 264
Piano box, 79
Personnel Administration (PA), 15, 36, 99
Plan data tab, 179
Planned work schedule infotype
PA0007, 446
Plus sign
Positioning of operations, 439
Positive time tracking, 203
Posting
AP, 121
Posting attributes, 314
Configuring, 299
Posting dates, 300
Post-payroll checks, 356
Post-payroll process, 29, 69
Post-tax deductions, 63, 84
Example, 84
Infotype, 63
Pre-payroll
Checks, 356
Process, 29
Pre-payroll data verification, 56
Tools and reports, 36
Pre-tax, 67, 83, 88, 186
Pre-tax deductions, 63, 84, 178, 245, 246, 282, 283, 307
Section 128, 181
Wage type, 193
PRINT, 444
Processing class, 67, 72, 79, 80, 81, 90, 164, 245, 299, 313, 427
5, 67
30, 439
50, 88, 366, 369
50 replacement, 86
59, 67, 228, 234
65, 67
65 pre-tax deductions, 98
66, 86, 137, 366
71, 246, 250, 264, 284, 285
Adjusting, 79
Commonly used, 72
Earning wage types, 82
P66, 137
Table, 246
Processing component
Third-party, 175
Production mode
Running Tax Reporter, 265
Proration, 140, 368
Provider, 176
Cost, 181
Provider contributions
Wage type, 193

Q

Quarterly report, 266

R

Rate, 69
Reciprocity, 275
Local tax authority, 243
Reconciliation run, 318
Recurring payments
Infotype, 245
Referral bonus, 51

Reimbursements, 284, 287
Remittance
Processing, 313
Rules, 226
Third-party, 68
Third-party due date, 317
To benefits providers, 197
Reporting error, 256
Reports
Annual, 268
Audit, 268
Payroll reconciliation, 267
Quarterly, 266
RESET, 444
Residence-related tax area, 248
Residence tax authority, 31, 43
Resource planning, 323
Retirement plan deductions
Annual limits, 186
Retirement savings plans, 183, 184
RETRO, 166, 444
Retroactive accounting, 28, 34, 35, 49, 137, 140, 267, 280, 282
Arrears, 106
Data change, 36
Definition, 34
Forced payroll, 35, 46
Payroll, 369, 372
RETRO calculation operation, 444
ROUND, 445
RPCALCU0, 125, 323
US Payroll driver, 147
RPCLMSU0, 282, 289
RPCPCC00, 121
RPDLGA20
Wage type utilization report, 233
RPUBTCU0, 367
RTE, 445
Wage type element, 69
Rules, 123, 129, 131, 140, 156
Best practices, 358
Running error-free, 170
Step-by-step process, 171
U011, 446
UAC0, 299
*UD**, 141*
UD11, 137, 141

Rules (Cont.)
UD21, 137, 141
UGARN, 142
UGDN, 142, 236
UNAM, 137
UPTX, 139
UW14, 135, 151
Writing in US Payroll, 156
X013, 136
X020, 136
X023, 137, 140
X024, 137, 141
X025, 137, 141
X040, 372
XPPF, 140, 368
XVAL, 368
ZRU1, 140, 159
Rules Editor, 157, 359, 439
Rule UGARN
Calculation of garnishments, 237
Running balance
Wage type for displaying, 86
Running payroll, 262

S

Salary increase
Infotype, 33
Sales commission, 71
SAP configuration
Golden Rule, 132
SAP documentation, 33, 95, 127
SAP Service Marketplace, 188
Sarbanes-Oxley Act (SOX), 355
Savings bond, 25, 67, 84, 154
Savings plan, 50, 154, 176, 183, 198, 369
Infotype, 245
Schema, 30, 64, 123, 139, 177, 282, 308, 316, 322
Best practices, 358
Defined, 127
U000, 299, 363
U200, 308
U250, 311
UDD0, 104
ULK9, 323

Schema (Cont.)
- *US Public Sector, 152*
- *XLR0, 364*

Schema Editor, 129, 145, 146, 359
Section 128
- *Pre-tax deductions, 181*

SETIN, 441
Sign reversal, 88
Social Security
- *Taxes, 91, 244*
- *Tax types, 93*

Specification
- *Processing class, 72*

Split, 89, 134, 139, 252
Split indicators
- *Eliminating, 441*

State tax
- *Wage types, 92*

State unemployment insurance, 243, 266
Statutory reporting, 153
Student tax treatment, 244
Subschema, 125, 130, 238, 245, 253, 299
- *Basic data processing, 134*
- *QPCD, 155*
- *UAC0, 302, 373*
- *UAL0, 140, 368*
- *UAP0, 139, 367*
- *UBD0, 363*
- *UBE1, 139, 369*
- *UBE2, 154, 369*
- *UDBS, 369*
- *UDD0, 140, 369*
- *UDNT, 372*
- *UEND, 143, 374*
- *UGRN, 142, 371*
- *UIN0, 363*
- *UMC0, 137, 262, 366*
- *UNA0, 372*
- *UNAO, 142*
- *UNN0, 142, 373*
- *UPNR, 153*
- *UPPT, 153*
- *UPR0, 364*
- *URR0, 372*
- *UT00, 135, 365*
- *UTBS, 369*
- *UTX0, 138, 367, 370*

Schema (Cont.)
- *XPDD, 370*

Subsequent maintenance node, 250
SUBWT, 445
Supplemental, 433
- *Rates, 75, 243, 245*

Support pack, 71
Symbolic account assignment, 315

T

Table
- *5U8A, 133*
- *5U8C, 133*
- *558A, 133*
- *ACCR, 302*
- *ADR, 135*
- *ARRRS, 138, 141, 151*
- *BT (bank table), 143*
- *C1, 137*
- *CRT, 143*
- *DDNTK, 138, 141, 370*
- *GRDOC, 142*
- *GRORD, 142*
- *GRREC, 142*
- *IT (input table), 135, 137, 138, 139, 140, 141, 142, 365*
- *NAME, 135*
- *PARX, 136*
- *RT, 289*
- *RT (results table), 139, 142, 143, 151*
- *T5U8C, 323*
- *T511K, 157, 186, 220, 445*
- *T511P, 186, 188, 191, 192*
- *T512W, 79, 181, 254, 299, 313*
- *T558C, 323*
- *T558x, 311*
- *T588B, 322*
- *TAXR, 135*
- *TCRT, 254*
- *V0, 139, 140*
- *WPBP, 135, 151*
- *ZL, 137, 209, 365*

Tax, 25, 89, 133
- *Area, 248, 252*
- *Class, 247, 250, 254*

Tax (Cont.)
Classification, 264
Data, 36
Deduction, 43, 62
Infotypes, 242
Levels, 248
Levy, 223
Medicare, 244
Non-resident, 244
Override, 243
Processing, 138, 151, 241
Rate, 92
Schemas, 43
Social Security, 91, 244
Student, 244
Type, 92, 249, 252
Unemployment, 249
Wage types, 43, 62, 90, 92, 241
Withholding, 249
Year, 84
Year-end adjustments, 255
Taxable
Base, 254
Gross, 254
Income, 255
Salary or earnings, 283, 433
Wages, 285
Tax and arrears data
Table T5U8C, 323
Tax authority, 43, 92, 135, 151, 243, 247, 248, 260, 323
Attaching a tax model, 251
Tax calculation
Post-tax deductions, 84
Tax combo 84
Withholding taxes, 53, 252
Taxed-when-earned calculation, 260, 261
Taxes and accounts posting
Type C, 287
Tax form groups, 269
Tax implications
Overpayments, 279
Tax model, 241, 247, 250, 254
Attaching, 251
Creating custom, 250
Processing class, 246
Tax Reporter, 31, 241, 255, 264, 265
Running in test mode, 256
Steps before running, 265
Tax update bulletin (TUB), 266
TCRT
Payroll table, 259
Technical wage types, 66, 72, 81, 82, 93, 379
Earnings, 82
Slash, 66, 72, 232
Termination
Employee, 281
Test mode, 255
Running Tax Reporter, 265
Third-party
Payroll systems, 306
Posting, 317
Processing component, 175
Remittance, 31, 68, 88, 312
Remittance due date, 317
Running process, 316
Time-based payment, 366
Time clocking, 202
Time cluster, 136
Time evaluation, 39, 42, 202
Time management, 135
Time-related processing, 365
Time wage types, 137, 203
Tips
Processing class, 246
TM00, 217
TM04, 219
Total benefits, 88
Total deductions, 88
Total gross, 66
Transaction
PA20, 36
PA30, 36
PAUX, 256
PAUY, 256, 258
PC_PAYRESULT, 100
PE01, 129, 363
PE02, 157
PE04, 166
PU03, 289
SE16, 356
SE38, 282
SM31, 79, 186, 234, 299

Transaction data, 45
Infotypes, 47
True cumulation, 101
TWEG, 261
Type C
Off-cycle payroll, 263

U

U000, 125
US Payroll schema, 125
UACGF function, 303
UBD0
Subschema for basic data processing, 134
UBE1
Benefits subschema, 245
UBE2
Benefits subschema, 245
UCLM
Claims schema, 282
UINO
Subschema for initializing payroll, 134
UMOD, 446
UNA0
Subschema for net calculations, 130
Unemployment tax, 249
Infotype 0209, 243
Union dues, 25, 44, 67
Garnishments, 104
Permissibility, 68
Unions, 24
United Way
Garnishments, 104
Model wage type, 71
U.S. modifiers, 446
US Payroll
Components, 30
Equation, 34
Process, 30
Schema, 125
Uniqueness, 24
U.S. public sector
Accruals, 26
U.S. resident tax area, 446
U.S. schema
Changing, 143
U.S. subschemas, 132, 133, 367
USTAX function, 253
U.S. tax year, 84
U.S. wage type
Catalog, 64
UT00
Subschema, gross compensation and time, 135
UTAXR, 446
Utilization report
Wage types, 97
UTX0
Subschema for tax calculations, 130

V

VAKEY, 446
VALEN, 446
Valuation, 82, 427
Valuation of wage types, 136
Variable assignment split, 441
Variable key
Length, 446
Position, 446
Vendor, 177
Posting, 317
Vendor 200025
Voluntary deduction, 37
Voluntary garnishments, 228
VWTCL, 447

W

W-2, 268
Audit report, 264
Test run, 288
W-4, 36, 46
Changes to data, 53
Data change, 356
Wage type, 31, 40
/5U0 gross tax deductions, 93

Wage type (Cont.)
/5U9, 262, 287
/101 total gross pay, 379
/117 pension plans, 81
401(k), 189
/403, 314
/404, 300
/559 net pay, 379
/559 transfer funds to bank account, 94
/560, 282
/561, 282
/561 clearing, 286
/563, 286
1001, 284
1002, 284
1200, 70
2417, 303
Assigning to an infotype, 99
BA13, 193
Base wage for BSI, 80
BE13, 193
BE14, 181
Benefits-related, 409
Best practices, 358
BP13, 193
BP14, 181
BR13, 193
BR14, 181
/BT1, 193
Catalog, 71
Categories, 68
Characteristics, 78
Commonly used technical wage types, 379
Configuring, 64
Cumulation, 439
Defined, 61, 64
Dropped, 235
Elements of, 69
Factoring, 81
/G00, 232, 236
/G01, 232
Groups, 77, 90
Lifecycle, 64, 65
M720 United Way, 71
Manipulation, 427
MFT1, 284

Wage type (Cont.)
MFT1 copying, 284
MG20, 232
MRP1, 285, 287, 289
MRP1 copying, 285
Numbering, 72
Payroll report, 100
Permissibility, 79
Posting, 109
Posting to expense accounts, 68
Printed or not printed, 68
Reporter, 143
Template, 375
Testing, 97
Time, 137
Utilization report, 97, 233
Valuation, 136
/X02, 369
/Z02, 369
Wage types
Starting with B, 66
Starting with M, 66
WCWP, 447
Weekly payroll, 126
WGTYP, 447
Withholding overrides, 243
Withholding tax, 249
Wage type, 89
Workplace and basic pay data
OUTWP, 443
Work-related tax area, 248
Work schedules, 203
Work tax, 43, 254
WPBP function, 134
Write-off, 284

X

X009, 446

Y

YANA, 261
Subtype of Infotype 0221, 259

YAWA, 262
Subtype of Infotype 0221, 259
Year-end
Adjustments, 137, 241, 255
Adjustment workbench, 241, 255, 257, 286, 379
Year-to-date, 64, 82, 88, 143
Cumulations, 31

Z

Z, 187
ZERO, 447
Zero-net-checks, 95
Zero net pay, 93, 95