

HR2005

MARCH 14 – 16 • ORLANDO

Produced and conducted by WIS, publisher of SAP Insider, with permission of SAP AG

Master the Six SAP Payroll “Building Blocks”: Features, Schemas, Rules, Operations, Operation Parameters, and Wage Type Integration

Cal Castleberry
RowSix LLC

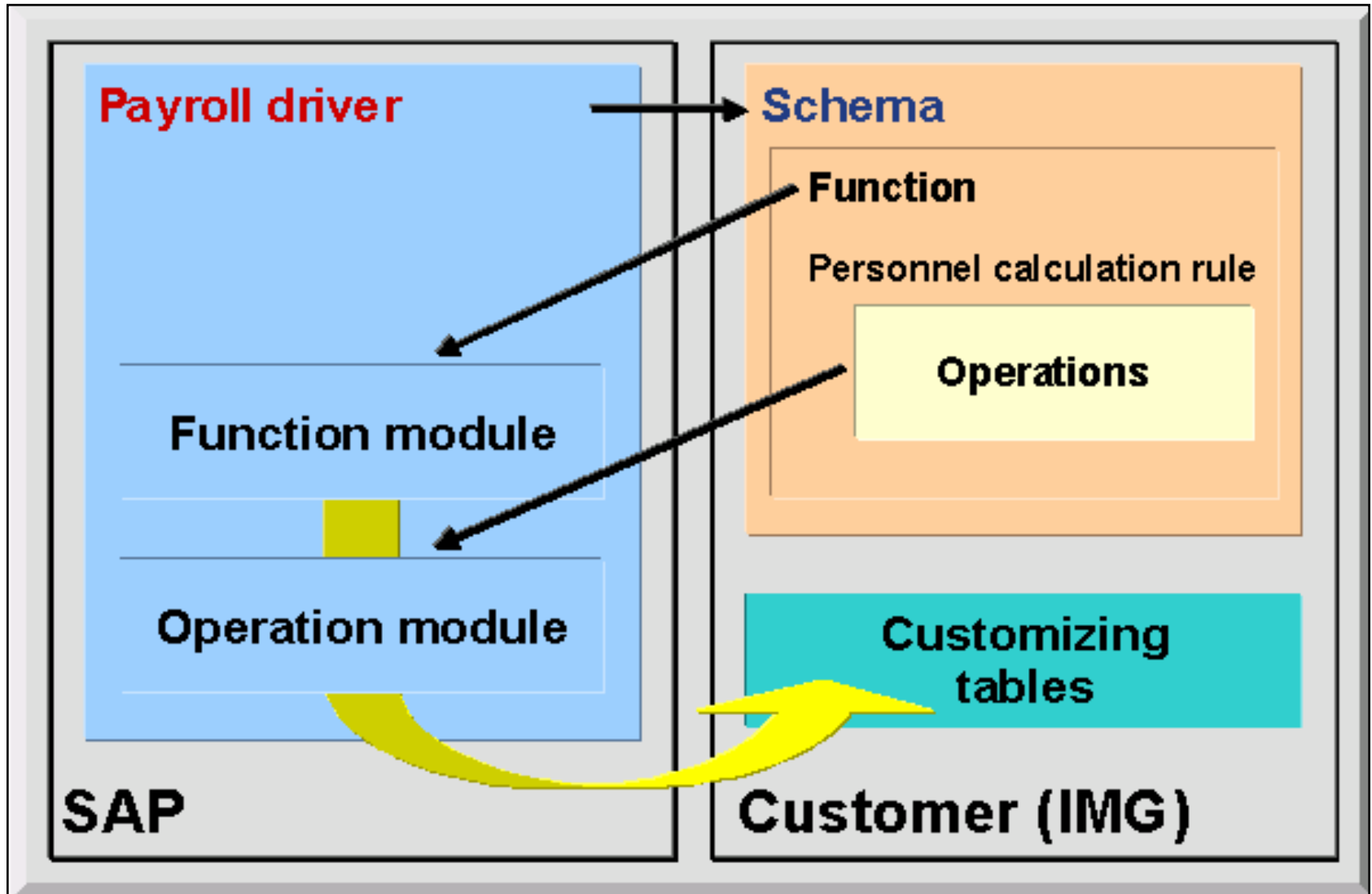
What We'll Cover ...

- Part of the “learning curve” associated with the use of SAP payroll accounting is the understanding of the underlying fundamental components. This session explains these fundamental components (the building blocks) in a way that allows them to be easily understood.
- This includes:
 - ♦ The HR component of features
 - ♦ The payroll components of the schema
 - ♦ The functions within the schema
 - ♦ The rules (cycles)
 - ♦ The operations within the rules
 - ♦ The parameters to the operations. We will cover what each component is and how each relates to the other, including the integration of wage type processing in payroll accounting

What We'll Cover ...

- HR/Payroll features (decision trees) – configuration
- Schemas and subschemas – configuration
- Payroll functions – SAP ABAP
- Calculation rules (or cycles) – configuration
- Operations within rules – SAP ABAP
 - ◆ Parameters used for operations – configuration
 - ◆ Parameters available for use – SAP ABAP
- Wage type integration – tie it together
- Wrap-up

What We'll Cover ... To the ABAP Code Level



What We'll Cover ...

- HR/Payroll features (decision trees) – configuration
- Schemas and subschemas – configuration
- Payroll functions – SAP ABAP
- Calculation rules (or cycles) – configuration
- Operations within rules – SAP ABAP
 - ◆ Parameters used for operations – configuration
 - ◆ Parameters available for use – SAP ABAP
- Wage type integration – tie it together
- Wrap-up

The Concept of HR/Payroll Features (Decision Trees)

- **What are HR/Payroll features?**
 - ♦ Features are decision trees designed to allow you to configure the logical determination of data entry values
 - ♦ Decision trees allow for a reduction and/or elimination of data entry errors and required “intelligent” key strokes
- **When and where are HR/Payroll features relevant?**
 - ♦ Features are programmatically (ABAP) executed at dialog entry time as the various types of master data information (infotypes) are created (not effective dated)
 - ♦ The features available to be used and configured for any particular infotype can easily be determined and verified through the use of the standard SAP “debugger” functionality
 - ♦ Not directly relevant to the gross-to-net payroll accounting, but indirectly via the stored master data

Example Maintain/Display HR/Payroll Feature – ABKRS

- Transaction PE03 is used to display and/or maintain (configure) the decision tree

Features: Initial Screen

Feature **ABKRS** ABKRS Default value for payroll accounting area

Subobjects

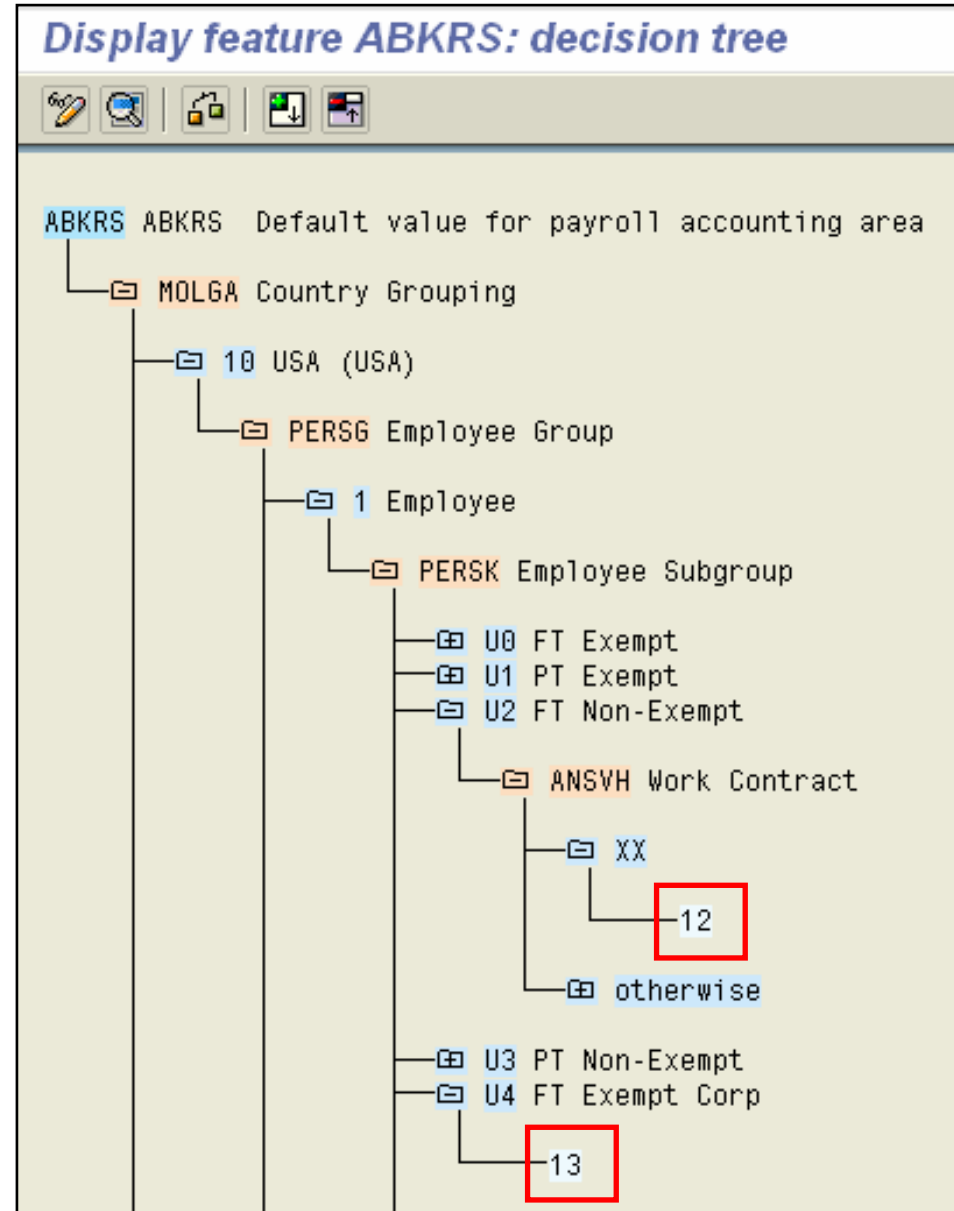
Decision tree
 Attributes
 Documentation

Display Change



Example HR/Payroll Feature (Decision Tree) – ABKRS

- In this example, the decision tree will default a payroll accounting area value of 12 (weekly) for the FT Non-Exempt employee
- A value of 13 (bi-weekly) will be defaulted for the FT Exempt Corp employee
- The features are configurable specific to customers' needs



When Do the Features (Decision Trees) Execute?

Create Organizational Assignment

Org Structure

Personnel No 10000100

Start 09/19/2004 to 12/31/2004

Enterprise structure

CoCode 100 INTERSTATE BRAN

Pers.area 4851 WHSO-Lorain,OH

Cost Ctr

Personnel structure

EE group 1 Employee

EE subgroup U2 FT Non-Exempt

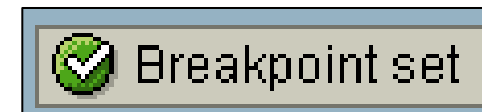
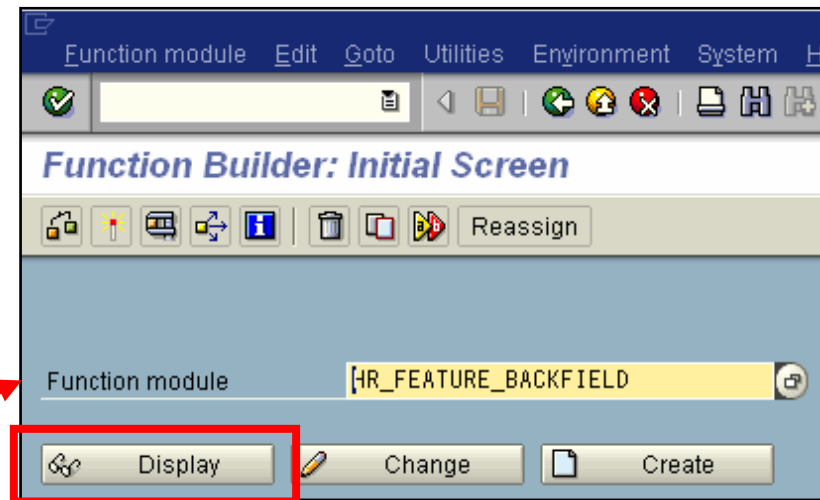
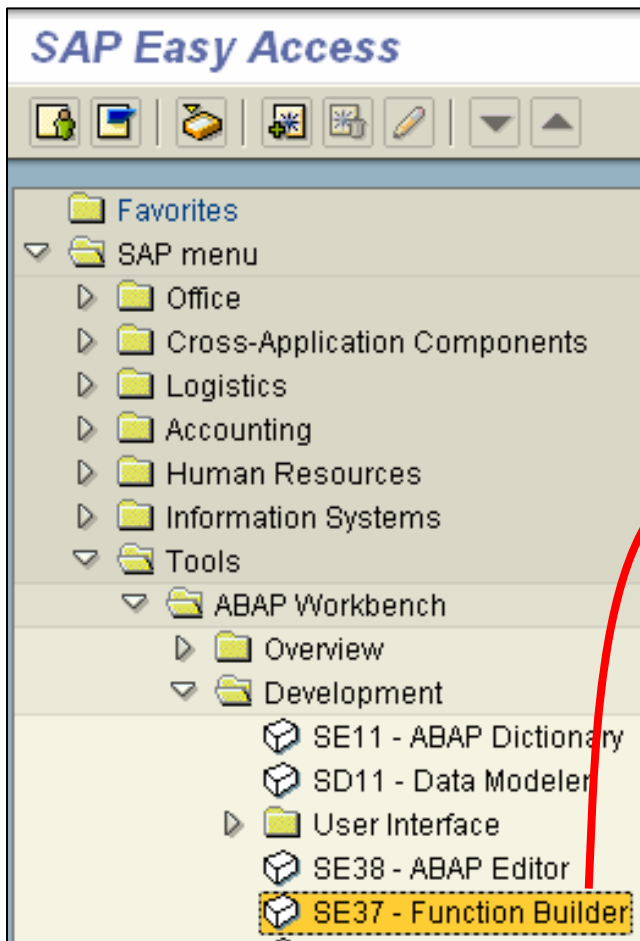
Payr.area 12 Weekly (Sun-Sat)

Contract

- In this example, the decision tree used the employee group value and the employee subgroup value to determine the correct payroll accounting area when the infotype 0001 (organizational assignment) was created. This results in a value of 12 (weekly) being defaulted for the payroll accounting area

Feature (Decision Trees) Determination and Verification Steps

- Prior to creating or changing an infotype, initiate the menu path or SE37 (function builder) transaction to display the HR_FEATURE_BACKFIELD function. Set a breakpoint.



Feature (Decision Trees) Determination and Verification Steps (cont.)

- Now, create the infotype that you would like to determine and verify the applicable features
- At each point that a feature is relevant, the debugger will stop

ABAP Debugger

Fields Table Breakpoints Watchpoints Calls Overview Settings

Main Program SAPLHRP3

Source code of LHRP3U01

29

FUNCTION HR_FEATURE_BACKFIELD

```
DATA: VALU VALUE 'V',           "Rückgabewert
      PROG VALUE 'P'.           "Programm-Name
DATA: PROGNAM(19).
* -----
* BACK = SPACE.                 "VLDN185055
* STATUS = 0.                    "Initialisierung "VLDAHRK047941
➔ STOP STATUS = '0'.            "VLDAHRK047941

SY-SUBRC = 0.
IF T549D-NAMEN NE FEATURE OR T549D-FUNID EQ SPACE. "QFLK11K104642
  SELECT SINGLE * FROM T549D WHERE NAMEN EQ FEATURE.
ENDIF.                               "!"
IF SY-SUBRC NE 0.
*-- Merkmal existiert nicht; d.h. es ist nicht generiert
  T549D-NAMEN = FEATURE.
* STATUS = 4.                    "VLDAHRK047941
  STATUS = '4'.                  "VLDAHRK047941
```

Field names: FEATURE ABKRS

Field contents:

Feature (Decision Trees) Determination and Verification Steps (cont.)

- Execute each line until you arrive at the PERFORM CALL_549B line



Execute (F8)

- Step into the PERFORM CALL_549B line

ABAP Debugger

Execute (F8)

Step Into (F5) | Table | Breakpoints | Watchpoints | Calls | Overview | Settings

Main Program: SAPLHRP3

Source code of: LHRP3U01

FUNCTION HR_FEATURE_BACKFIELD

```
ELSE.  
  PROGNAM = T549D-FUNCT.  
  PERFORM CALL_549B IN PROGRAM (PROGNAM)  
    USING BACK  
    STATUS  
    STRUC_CONTENT.
```

Secret

" VLDAHRK006111

Feature (Decision Trees) Determination and Verification Steps (cont.)

- Now, you can examine and verify the exact feature decision tree. Use the step into icon to verify the feature (decision tree) logic. In my example, the case statements that arrive to a BACK field value of 12 are indicated by the red arrows

```
ABAP Debugger
Fields Table Breakpoints Watchpoints Calls Over
Main Program /1PAPA/FEAT011ABKRS
Source code of /1PAPA/FEAT011ABKRS
FORM CALL_549B
FORM CALL_549B USING BACK STATUS STRUC STRUCTURE PME04.
SET EXTENDED CHECK OFF.
CASE STRUC-MOLGA .
  WHEN '10' .
CASE STRUC-PERSG .
  WHEN '1' .
CASE STRUC-PERSK .
  WHEN 'U0' .
  BACK = '12' .
  WHEN 'U1' .
  BACK = '12' .
  WHEN 'U2' .
CASE STRUC-ANSVH .
  WHEN 'XX' .
  BACK = '12' .
  WHEN OTHERS .
  BACK = '12' .
```

What We'll Cover ...

- HR/Payroll features (decision trees) – configuration
- Schemas and subschemas – configuration
- Payroll functions – SAP ABAP
- Calculation rules (or cycles) – configuration
- Operations within rules – SAP ABAP
 - ◆ Parameters used for operations – configuration
 - ◆ Parameters available for use – SAP ABAP
- Wage type integration – tie it together
- Wrap-up

The Concept of Schemas – Payroll Flow Control

- **What are schemas and subschemas?**
 - ♦ The schema controls the flow of payroll accounting
 - ♦ The schema is organized in a logical manner to easily reflect the normal flow of gross to net, and the recursive looping required in the case of a zero net check
 - ♦ The schema is configurable to allow for the flexibility of your business needs and requirements.
- **What is the RPCALCx0 program?**
 - ♦ The RPCALCx0 program is not a top-down program, but a warehouse of code to be chosen to be executed by the schema

Example Maintain/Display Schema – Payroll Flow Control

- Transaction PE01 is used to display and/or maintain (configure) the schemas and subschemas

Personnel Calculation Schemas: Initial Screen

Schema sic data (US)

Create

Subobjects

Source text
 Attributes
 Documentation

Display Change





Example Maintain/Display Schema – Payroll Flow Control (cont.)

1. Read basic data
2. Read payroll accounts from the payroll period last accounted
3. Time data processing
 - Read time data
 - Generate time wage types
 - Valuate time wage types
4. Import additional payments/deductions
5. Factoring
 - Reduce basic pay
 - Determine wage types for FI/CO
6. Statutory payments/deductions
7. Net payments/deductions
8. Determine payment amount

Example Maintain/Display Schema – Payroll Flow Control (cont.)

- In my example, the main payroll gross-to-net schema is Z000
- This schema determines the sub-schemas that will be used to accomplish the gross-to-net payroll accounting
- The schema is configurable

Edit Schema: Z000

Cmmnd Stack

Line	Func.	Par1	Par2	Par3	Par4	D	Text
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	ZT00					CDC - Gross compensation and time eval
000070	COPY	UREI				*	Travel expense
000080	BLOCK	BEG					Gross cumulation and tax processing
000090	IF		NAMC				if non-authorized manual check (*)
000100	COPY	ZMC0					CDC - Process Non authorized check (*)
000110	ELSE						else if non authorized manual check (*)
000120	COPY	ZAP0					CDC - Process add. pmts and deds
000130	COPY	ZAL0					CDC - Proration and cumulation gross
000140	COPY	UTBS					Save tables for iteration
000150	LPBEG						Begin of iteration
000160	COPY	UTBL					Load saved tables
000170	COPY	UDD0					Process deductions, Benefits
000180	COPY	ZTX0					CDC - Calculate taxes
000190	COPY	UXD1					Fill Currency fields
000200	COPY	UGRN					Calculate garnishments
000210	COPY	UNA0					Calculate net
000220	COPY	UDNT					Deductions not taken during loop ?
000230	LPEND						End of iteration
000240	ENDIF						to: if non authorized manual check (*)

What Are the Various Types of Standard Schemas?

- Following is a list of the SAP standard schemas supplied for various functional reasons. Regardless of the specific schema, each controls the flow of programming to be selected from the warehouse of code stored in the RPCALCU0 program

Schema Directory

Schema	Ver	Text	Selected.....
U000	MOD	US accounting schema for RPCALCU0	
U001	STD	ADP schema for RPCALCU0	
U00C	STD	US accounting schema for RPCALCU0 CE	
U200	STD	Generic Outsourcing Schema - U.S.	
U250	STD	Transfer of Imported Payroll Results 558d/5u8c/558e (US-Specific)	
U500	MOD	Evaluation run - US	
U50C	STD	Evaluation run - US CE	

- You can copy and enhance each of these executable schemas via configuration to meet specific requirements

Example Maintain/Display Schema – Payroll Flow Control

- For example, you may not wish to implement Travel Management at this point in time
- Or, you may require enhancements to the way additional payments and deductions are processed
- Or you may desire tax posting enhancements to allow the use of auto-clearing functionality in FI/CO

Edit Schema: Z000



Cmmnd Stack

Line	Func.	Par1	Par2	Par3	Par4	D	Text
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	ZT00					CDC - Gross compensation and time eval
000070	COPY	UREI				*	Travel expense
000080	BLOCK	BEG					Gross cumulation and tax processing
000090	IF		NAMC				if non-authorized manual check (*)
000100	COPY	ZMC0					CDC - Process Non authorized check (*)
000110	ELSE						else if non authorized manual check (*)
000120	COPY	ZAP0					CDC - Process add. pmts and deds
000130	COPY	ZAL0					CDC - Proration and cumulation gross
000140	COPY	UTBS					Save tables for iteration
000150	LPBEG						Begin of iteration
000160	COPY	UTBL					Load saved tables
000170	COPY	UDD0					Process deductions, Benefits
000180	COPY	ZTX0					CDC - Calculate taxes
000190	COPY	UXD1					Fill Currency fields
000200	COPY	UGRN					Calculate garnishments
000210	COPY	UNA0					Calculate net
000220	COPY	UDNT					Deductions not taken during loop ?
000230	LPEND						End of iteration
000240	ENDIF						to: if non authorized manual check (*)

Example Subschema UBD0 – Basic Data Flow Control

- In this example, the UBD0 subschema will control the flow of payroll accounting based on SAP-supplied logic

Edit Schema: UBD0

Cmmnd | Stack

Line	Func.	Par1	Par2	Par3	Par4	D	Text
000010	BLOCK	BEG					Basic data processing
000020	ENAME						Retrieve employee name
000030	WPBP						Read org. assignment / basic pay
000040	P0002						Read personal data
000050	P0006						Read address
000060	P0207	2					Read tax data
000070	GON						Continue with complete data
000080	P0014	UW14	GEN	NOAB			Split WPBP and set APZNR for P0014
000090	PRINT	NP	NAM				Print employee name
000100	PRINT		WPBP				Print org. assignment / basic pay
000110	PRINT		TAXR				Print resident authority
000120	PRINT		PERM				Print personal data
000130	IF		SPRN				If special run
000140	RFRSH		IT				Clear internal table IT
000150	ENDIF						Endif
000160	BLOCK	END					

Example Maintain/Display Subschema – Payroll Flow Control

- Or, you may wish to enhance the SAP-supplied subschema payroll flow control
- For example, you may not be implementing integrated SAP time management functionality
- Or you may have unique rules regarding the way you value shift differential rates for various union contracts
- The subschema is configurable

Edit Schema: ZT00

Cmmnd | Stack

Line	Func.	Par1	Par2	Par3	Par4	D	Text
000290	PRINT	NP	IT				Print IT
000300	P2010	Z930	GEN	NOAB			CDC - Process employee remuneration info
000310	UTIPS						Tip processing
000320	PIT	UTAL	P84	NOAB			Total tip allocation amounts
000330	PALP	X012	GEN				Val. bases for different payments
000340	PRINT	NP	ALP				Table of different payments
000350	PIT	XALP		NOAB			Increased val.basis+extra pay+premium
000360	IF	UTRR					* If workweek and hourly paid
000370	COPY	UTR0					* FLSA: overtime valuation with reg.date
000380	ELSE						* Else usual valuation of time wage types
000390	ZLIT				ASXN	*	Time wage types in IT
000400	ZLIT				MYCR	*	Time wage types in IT
000410	P0416						Time Quota Compensation
000420	PIT	ZSD1					CDC - Det Shift Dif modif by wage type
000430	PIT	X015	GEN	NOAB			Valuation of time wage types
000440	COPY	ZUS0					Convert amts down to 2 decimals
000450	IF	UTRR					If workweek and hourly paid
000460	COPY	ZTR0					CDC - FLSA: OT valuation with reg.date
000470	ELSE						Else usual valuation of time wage types
000480	ENDIF						Endif: workweek and hourly paid
000490	PIT	X009	GEN	NOAB			Remove val. bases with ALP split
000500	COPY	XIW0				*	Incentive wages
000510	PIT	Z020	P03				CDC - Gross & RT storage time wage types
000520	ENDIF						Endif: Special Payroll Run (IF SPRN)
000530	BLOCK	END					

What We'll Cover ...

- HR/Payroll features (decision trees) – configuration
- Schemas and subschemas – configuration
- Payroll functions – SAP ABAP
- Calculation rules (or cycles) – configuration
- Operations within rules – SAP ABAP
 - ◆ Parameters used for operations – configuration
 - ◆ Parameters available for use – SAP ABAP
- Wage type integration – tie it together
- Wrap-up

The Concept of Payroll Functions (ABAP Code)

- **What are payroll functions?**
 - ♦ The function is a logical block of ABAP code that is contained in the RPCALCU0 warehouse of code
- **When and where are payroll functions relevant?**
 - ♦ The use of a functions is configured in the schema and subschemas and executed at payroll processing time
 - ♦ We will learn how to find and research the ABAP code associated with any payroll function
 - ♦ The use of a parameter for a function is configured in the schema and subschema
 - ♦ We will learn how to determine which parameters are available for use with any payroll function

Example Subschema UBD0 – Basic Data Flow Control

- In this example, the UBD0 subschema will control the flow of payroll accounting based on the use of SAP-supplied functions

Edit Schema: UBD0


Cmmnd | Stack

Line	Func.	Par1	Par2	Par3	Par4	D	Text
000010	BLOCK	BEG					Basic data processing
000020	ENAME						Retrieve employee name
000030	WPBP						Read org. assignment / basic pay
000040	P0002						Read personal data
000050	P0006						Read address
000060	P0207	2					Read tax data
000070	GON						Continue with complete data
000080	P0014	UW14	GEN	NOAB			Split WPBP and set APZNR for P0014
000090	PRINT	NP	NAM				Print employee name
000100	PRINT		WPBP				Print org. assignment / basic pay
000110	PRINT		TAXR				Print resident authority
000120	PRINT		PERM				Print personal data
000130	IF		SPRN				If special run
000140	RFRSH		IT				Clear internal table IT
000150	ENDIF						Endif
000160	BLOCK	END					

Help Specific to a Function (WPBP) and Parameters

- By placing your cursor next to any of the functions and using the Help key (F1), you can learn about the various parameters available for use (configuration) with this SAP-supplied function

Documentation Import work center and basic pay data (EN)

Link 

Import Work Center and Basic Pay Data

Object

Function

Use

Function WPBP transfers master data on the work center and basic pay for the payroll period to internal table WPBP. The basic pay wage types are stored in the internal table IT.

Procedure

This data usually remains the same for the entire period. However, in certain cases different data may exist for different partial periods. If this is the case, the entries are identified in internal tables WPBP and IT, by means of split indicators.

Syntax

Par2	different pay-scale type No check. All entered values are valid.
Par3	rel. time unit WPBP for each time unit

Example Maintain/Display Function – WPBP

- Transaction PE04 is used to display, maintain, or create functions

Maintain Functions and Operations

Activate all Input parameters Output parameters

Name

Object class
 Payroll
 Time management

Object type
 Function
 Operation

Create Change Display Delete

Change



Example Maintain/Display Function – WPBP (cont.)

- Here you may display or maintain (configure) the characteristics of a function such as the country-specific use and parameters available for use by the function
- You may create your own functions
- The standard ABAP name begins with FU

Characteristics Function Payroll WPBP

Infotypes Grouping Input parameters Output parameters

Function WPBP Import work center and basic pay data

Name of form routine

Standard name FUWPBP

Self-defined

Country assignment

Eligi...	Country Grouping
<input checked="" type="checkbox"/>	Argentina
<input checked="" type="checkbox"/>	Australia
<input checked="" type="checkbox"/>	Austria
<input checked="" type="checkbox"/>	Belgium
<input checked="" type="checkbox"/>	Brazil
<input checked="" type="checkbox"/>	Canada
<input checked="" type="checkbox"/>	China

Parameter list

Number	Parameter	Meaning	Ty.	Mandat	Relev
2	ALTTA	Alt. pay scale type	A	<input type="checkbox"/>	<input type="checkbox"/>
3	SUNIT	rel. time unit		<input type="checkbox"/>	<input type="checkbox"/>
4	SUBTY	IT0008 subtype	A	<input type="checkbox"/>	<input type="checkbox"/>

When Are the Functions (ABAP Code) Executed?

Payroll Driver (USA)

Selections from Search helps

Payroll period

Payroll area 12 09/19/2004 To

Current period

Other period 17 2004

Selection

Personnel number 10000099

Payroll area 12

General program control

Reason for payroll

Off-cycle payroll

Schema Z000

Forced retro.accounting as of

Test run (no update)

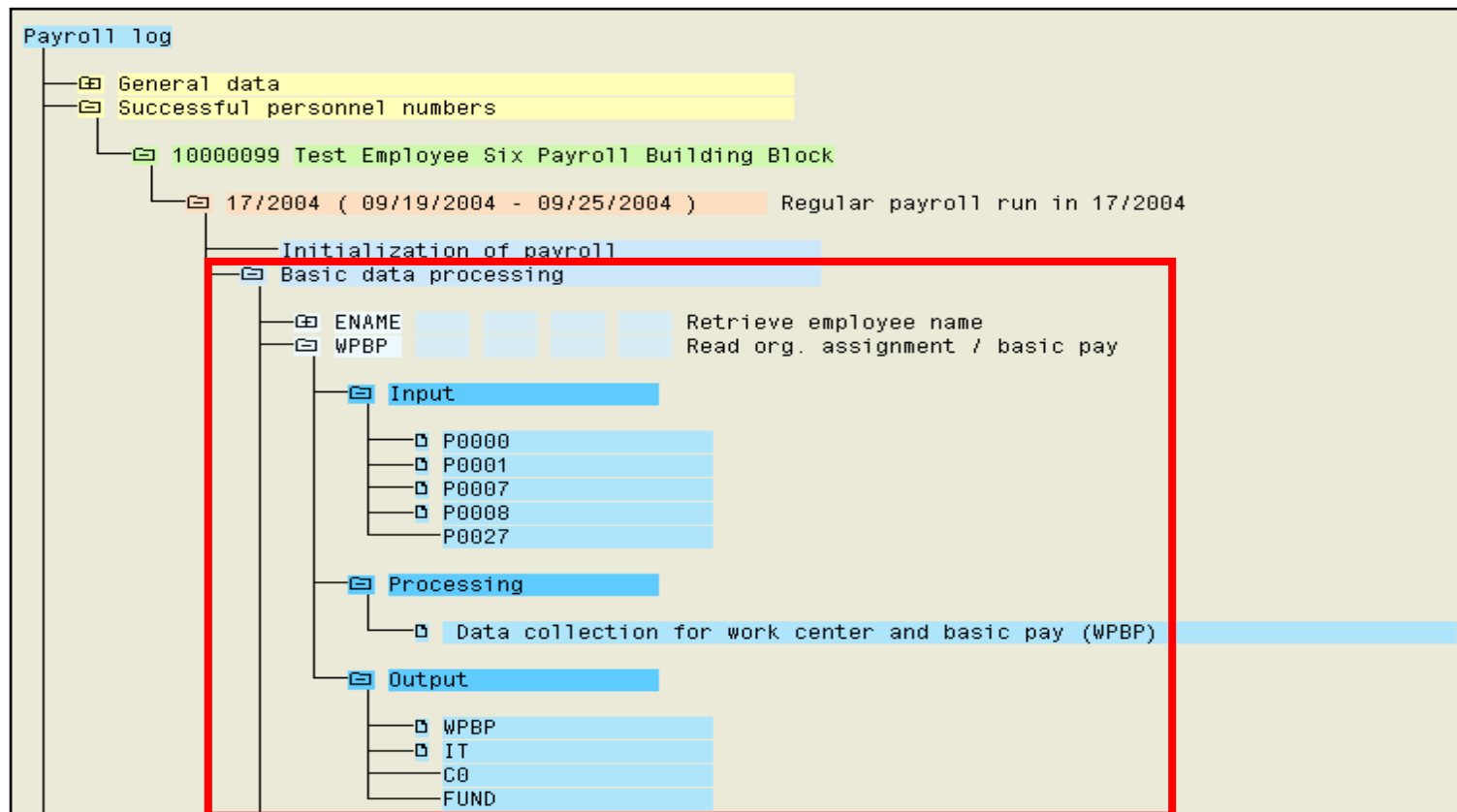
Log

Display log

- In this example, the executable schema Z000 will execute the subschemas UDB0, which will execute the function WPBP
- The input, processing, and output resulting from the execution of any function can be displayed by turning on the Display log functionality

Functions (ABAP Code) Determination and Verification

- The following payroll log information allows for the analysis of the payroll subschema UBD0 and the function inputs, the processing, and the resulting outputs



Functions (ABAP Code) Determination and Verification (cont.)

- To determine and understand the ABAP code that is being executed for any function, first you must enter the debugger mode using the '/h' transaction. Then execute payroll, and set a breakpoint at the FU function.



Functions (ABAP Code) Determination and Verification (cont.)

The screenshot shows the ABAP Debugger interface. The 'Create Breakpoint' dialog is open, showing the following fields:

Breakpoint at	subroutine/method/module (processing block)
Program	RPCALCU0
Subroutine/Method/Module	FUWPBP

A red box highlights the 'Program' and 'Subroutine/Method/Module' fields. A red arrow points from the 'Function module...' option in the main menu to the 'Program' field. A 'Breakpoint set' confirmation box is visible at the bottom right.

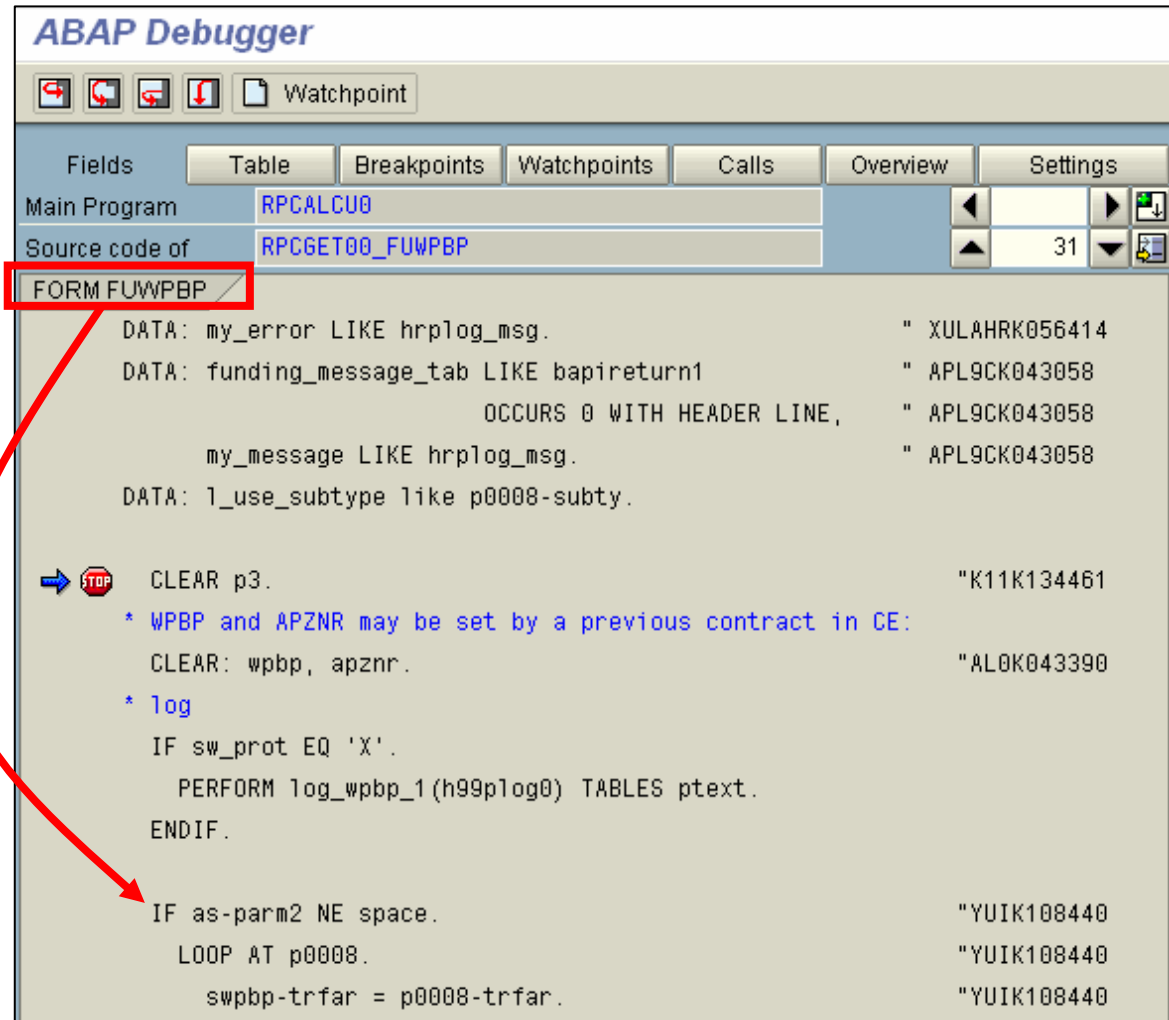


Functions (ABAP Code) Determination and Verification (cont.)

- “Run” and you arrive at the FORM FUWPBP ABAP code



- You can verify and analyze the availability and use of parameters to configure your use of the ABAP function via standard debugger functionality



The screenshot shows the ABAP Debugger interface. The title bar reads "ABAP Debugger". Below the title bar are several icons and a "Watchpoint" button. A menu bar contains "Fields", "Table", "Breakpoints", "Watchpoints", "Calls", "Overview", and "Settings". The "Main Program" field shows "RPCALCU0" and the "Source code of" field shows "RPCGET00_FUWPBP". A red box highlights the "FORM FUWPBP" entry in the source code list. The main area displays the source code for this form, including data declarations and logic. A red arrow points from the "FORM FUWPBP" entry to the code. The code includes comments like "* WPBP and APZNR may be set by a previous contract in CE:" and logic for clearing variables and logging.

```
DATA: my_error LIKE hrplog_msg.                " XULAHRK056414
DATA: funding_message_tab LIKE bapireturn1     " APL9CK043058
      OCCURS 0 WITH HEADER LINE,              " APL9CK043058
      my_message LIKE hrplog_msg.              " APL9CK043058
DATA: l_use_subtype like p0008-subty.

➔ CLEAR p3.                                    "K11K134461
   * WPBP and APZNR may be set by a previous contract in CE:
   CLEAR: wpbp, apznr.                          "AL0K043390
   * log
   IF sw_prot EQ 'X'.
     PERFORM log_wpbp_1(h99plog0) TABLES ptext.
   ENDIF.

   IF as-parm2 NE space.
     LOOP AT p0008.
       swpbp-trfar = p0008-trfar.                "YUIK108440
                                           "YUIK108440
                                           "YUIK108440
```

What We'll Cover ...

- HR/Payroll features (decision trees) – configuration
- Schemas and subschemas – configuration
- Payroll functions – SAP ABAP
- Calculation rules (or cycles) – configuration
- Operations within rules – SAP ABAP
 - ◆ Parameters used for operations – configuration
 - ◆ Parameters available for use – SAP ABAP
- Wage type integration – tie it together
- Wrap-up

The Concept of Payroll Calculation Rules (Configuration)

- **What are payroll rules?**
 - ♦ Standard calculation rules are provided by SAP as a baseline for gross-to-net payroll accounting
 - ♦ Rules are configurable, and customer-specific rules can easily be created to solve complex business requirements and problems
 - ♦ The calculation rules can be wage type specific or generic
 - ♦ The calculation rules can be based on the type of employee, salary exempt, salary non-exempt, hourly non-exempt, etc.
 - ♦ The SAP calculation rules (cycles) are your friend and the technical strength of flexibility

The Concept of Payroll Calculation Rules (Configuration) (cont.)

- **When and where are payroll rules relevant?**
 - ♦ The use of a calculation rule is configured in the schema and sub-schemas and executed at payroll processing time
 - ♦ We will learn how to find and research the payroll calculation rules associated with any payroll function

Example Subschema UT00/ZT00 (Time Data Processing)

- In this example, this portion of the ZT00 subschema will execute the process input table (PIT) payroll function (ABAP) using the calculation rule of Z020 (first parameter to the function)

Edit Schema: ZT00

Cmmnd Stack

Line	Func.	Par1	Par2	Par3	Par4	D	Text
000510	PIT	Z020	P03				CDC - Gross & RT storage time wage types
000520	ENDIF						Endif Special Payroll Fun (F. CDCN)
000530	BLOCK	END					

FORM FUPIT

* DASSELBE FORMAT WIE DIE EINGABETABELLE HAT. SIE WIRD ZUR EINGABE-
* TABELLE DER NAECHSTEN PHASE.
*
➔ ccycl = as-parm1.
* perform phase-heading. "XDOALRK030399
LOOP AT it.

as-parm1	Z020
----------	------

Help Specific to a Function (PIT) and Parameters

- By placing your cursor next to any of the functions and using the Help key (F1), you can learn about the parameters available for use (configuration) with this SAP-supplied function

Process Input Table

Acronym: PIT (Process Input Table)

Object

Function

Use

Function PIT calls a personnel calculation rule which processes the contents of the IT according to designated rules.

Procedure

Each wage/salary type from the IT is processed according to the PC rules (in Par1). PC rules generally change the wage/salary type (for example, valuation, cumulation) and store the changed wage/salary type in an output table (Operation ADDWT).

Syntax

Par1	nnnn	Personnel calculation rule Table T52CE
Par2	GEN	Type of call Generic call of PC rule The PC rule is called independent of the wage/salary type.
	*	Effect See Par2 (GEN)
	Pnn	Processing class The PC rule carries out a series of operations for each wage/salary type and for each specification of a processing class (nn). Wage type valuation table (T512W)

Example Maintain/Display Rule – Z020 'Store Gross Amt'

- Transaction PE02 is used to display, maintain, or create rules

Personnel Calculation Rules : Initial Screen

Rule Cumulation of gross amount

Subobjects

Source text

 ESGrp grouping

 Wage/time type

Attribute

Documentation



Example Maintain/Display Rule – Z020 ‘Store Gross Amt’ (cont.)

- Here you may display and/or maintain (configure) a payroll calculation rule
- You may determine specific types of employees and/or wage types for which this calculation rule applies

Display calculation rule: Z020

Z020 Cumulation of gross amount

ESGrp grouping	*
Wage/time type	****

VWTCL 03 Processing class

ERROR	Cancel processing
ADDWT *	OT Output table
1	
ADDWTE*	RT Results table
ELIMI KTX	Elim.time period ID
ADDCU	Cumul.WT in TAB.OT

Example Maintain/Display Rule – Z020 ‘Store Gross Amt’ (cont.)

- You may want to evaluate a wage type attribute (processing class value)
- You can configure which operations should be performed for each situation

Display calculation rule: Z020

ESGrp grouping *

Wage/time type ****

Z020 Cumulation of gross amount

- *
 - ****
 - VWTCL 03 Processing class
 - *
 - ERROR Cancel processing
 - 0
 - ADDWT * OT Output table
 - 1
 - ADDWTE* RT Results table
 - ELIMI KTX Elim.time period ID
 - ADDCU Cumul.WT in TAB.OT

When Is the Calculation Rule (Configuration) Executed?

Payroll Driver (USA)

Selections from Search helps

Payroll period

Payroll area 12 09/19/2004

Current period

Other period 17 2004

Selection

Personnel number 10000099

Payroll area 12

General program control

Reason for payroll

Off-cycle payroll

Schema Z000

Forced retro.accounting as of

Test run (no update)

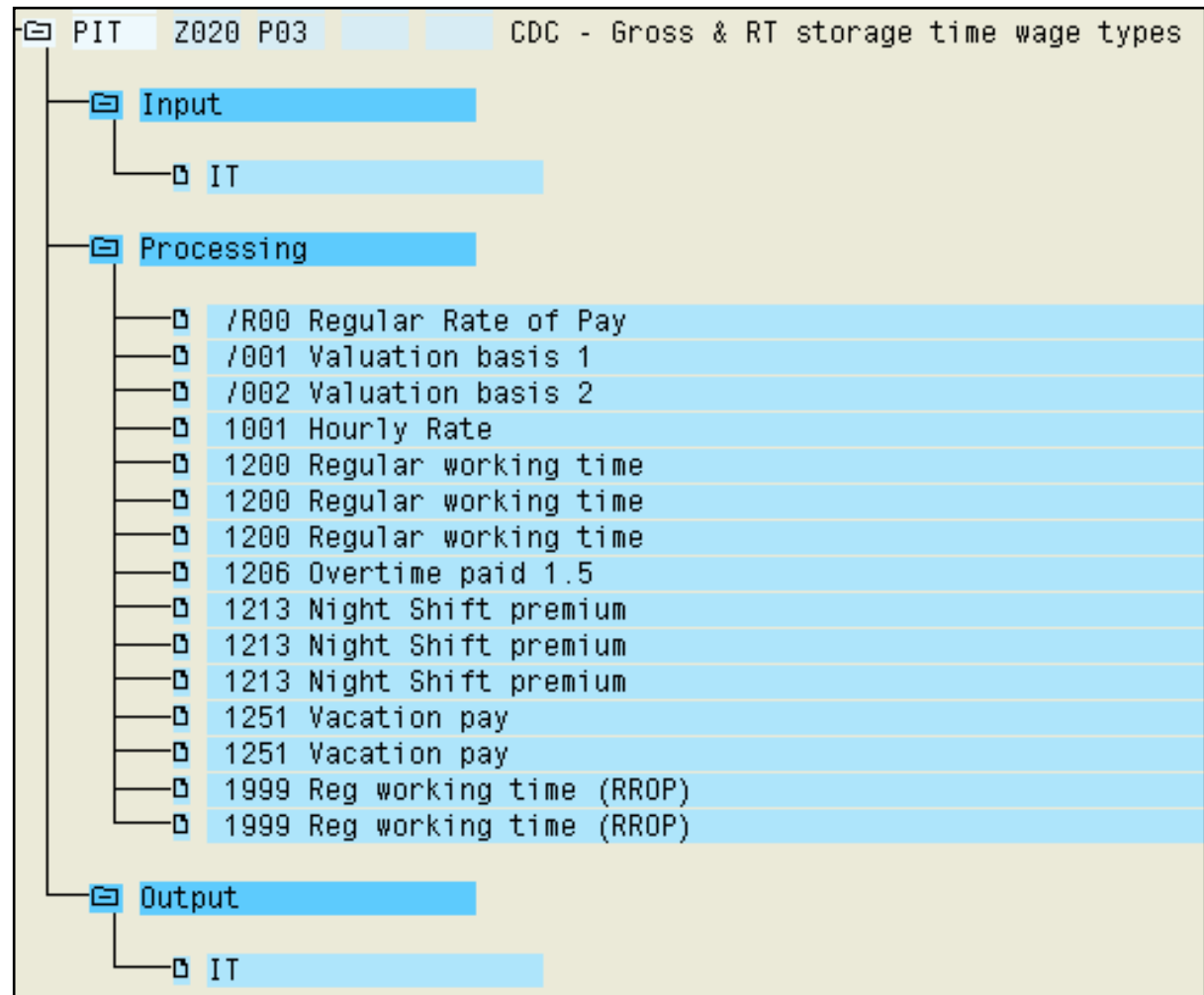
Log

Display log

- In this example, the executable schema Z000 will execute the sub-schema ZT00, which will execute the function PIT, which will execute the calculation rule Z020 (store gross amount)
- You display the input, processing, and output resulting from the execution of any function via the Display log functionality

Calculation Rules Determination and Verification

- The following payroll log information allows for the analysis of the payroll subschema ZT00, the function PIT, and the calculation rule Z020 inputs, the processing, and the resulting outputs...



Calculation Rules Determination and Verification (cont.)

- To determine and understand the ABAP code that is being executed for any specific calculation rule, first you must set a break in the subschema prior to the calculation rule, execute payroll, and then set a breakpoint at the FUPIT function, as shown in the next slide
- The setting of a break in the schema and/or subschema, which will be specific to a user, requires the creation of an AB4 parameter. This is created via the path of: System > User Profile > Own Data > Parameters > Enter Parameter ID of 'AB4' and a 2-byte parameter value

Parameter ID	Parameter value	Short Description
AB4	CC	ABAP prefix

Calculation Rules Determination and Verification (cont.)

- Setting a breakpoint (cont.)

000510	BREAK	CC				CDC - Tesing Break prior to Z020
000520	PIT	Z020	P03			CDC - Gross & RT storage time wage types

The screenshot shows the ABAP Debugger interface. The 'Breakpoints' menu is open, showing options like 'Save', 'Breakpoint at', 'Create/delete', 'Delete all', 'Deactivate/Activate', 'Deactivate all', and 'Activate all'. The 'Create Breakpoint' dialog box is open, showing the 'Program' field set to 'RPCALCU0' and the 'Subroutine/Method/Module' field set to 'FUPIT'. A red arrow points from the 'PIT' entry in the table above to the 'FUPIT' entry in the dialog box. Another red arrow points from the 'Breakpoint at' menu item to the 'Create Breakpoint' dialog box. A third red arrow points from the 'FUPIT' entry in the dialog box to the 'Breakpoint set' confirmation message.



Calculation Rules Determination and Verification (cont.)

- “Run” and you arrive at the FORM FUPIT ABAP code specific to the calculation rule Z020 (store gross amount)



Run (to Cursor) (F8)

- You can verify and analyze the processing of your configured calculation rule via the standard debugger functionality

ABAP Debugger

Fields Table Breakpoints Watchpoints Calls Overview Settings

Main Program RPCALCU0

Source code of RPCMAS09_FUPIT

FORM FUPIT

```
* DASSELBE FORMAT WIE DIE EINGABETABELLE HAT. SIE WIRD ZUR EINGABE-
* TABELLE DER NAECHSTEN PHASE.
-----
-> STOP ccycl = as-parm1.
* perform phase-heading. "XDOALRK030399
LOOP AT it.
  plog4_perform plog_header_cycle(h99plog0)
    using it-lgart calcmolga. "XDOALRK000847
  MOVE ccycl TO i52c5.
  MOVE-CORRESPONDING it TO i52c5.
  ot = it.
  PERFORM regel.
  plog1_perform plog_check_rule_performed(h99plog0). "XDOALRK000847
ENDLOOP.
REFRESH it.
PERFORM ot-in-it-append. "append statt collect, beinhaltet refresh ot.
ENDFORM. "END OF FUPIT
```

Field names 1 - 4 Field contents

as-parm1	Z020
----------	------

What We'll Cover ...

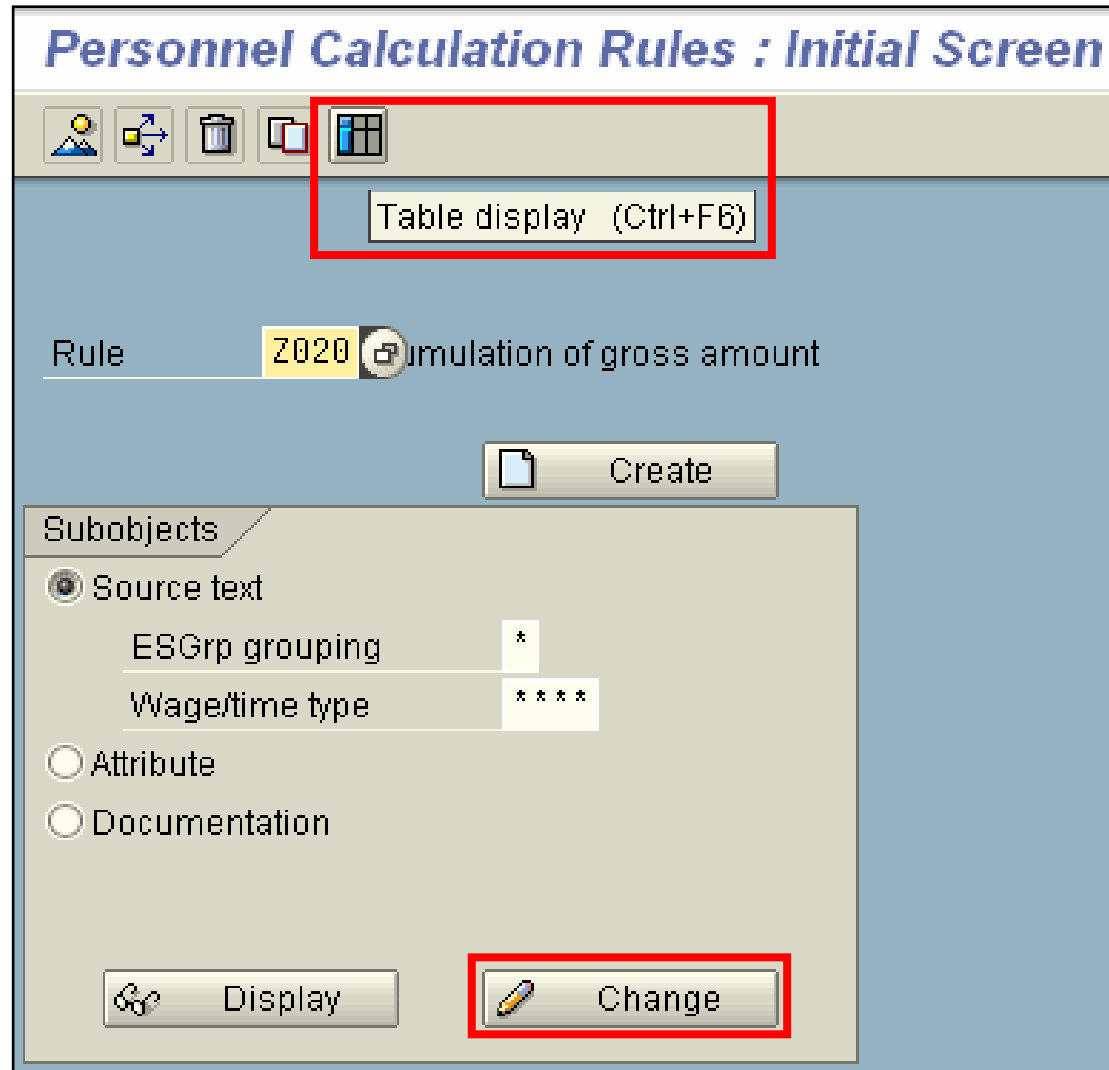
- HR/Payroll features (decision trees) – configuration
- Schemas and subschemas – configuration
- Payroll functions – SAP ABAP
- Calculation rules (or cycles) – configuration
- Operations within rules – SAP ABAP
 - ◆ Parameters used for operations – configuration
 - ◆ Parameters available for use – SAP ABAP
- Wage type integration – tie it together
- Wrap-up

The Concept of Calculation Rule Operations (ABAP)

- **What are calculation rule operations?**
 - ♦ The operation is a logical block of ABAP code that is contained in the RPCALCU0 warehouse of code
- **When and where are calculation rule operations relevant?**
 - ♦ The use of an operation is configured in the payroll calculation rule and executed at payroll processing time
 - ♦ We will learn how to find and research the ABAP code associated with any payroll operation
 - ♦ The use of a parameter for an operation is configured in the payroll calculation rule
 - ♦ We will learn how to determine which parameters are available for use with any payroll operation

Example Maintain/Display Rule – Z020 ‘Store Gross Amt’



- Transaction PE02 is used to display, maintain, or create rules



Example Maintain/Display Operation (ABAP) – ADDWT

- Here you may display and/or maintain (configure) the use of an operation (ABAP)

*Edit Rule: Z020 ES Grouping * Wage Type/Time Type *****


Cmmnd |

Line	Var.Key	CL	T	Operation	Operation	Operation	Operation	Operation	Operation *
000010			D	VWTCL 03					
000020	*			ERROR					
000030	0			ADDWT *					
000040	1			ADDWTE*	ELIMI KTX	ADDCU			
000050	2			ELIMI *	RESET AR	ADDWTE*			
000060	3			ADDWTE*	ELIMI KTX	ADDCU			
000070	4			ELIMI *	RESET AR	ADDWTE*			
000080	5			ADDWTE*	ELIMI KTX	ADDCU	NEXTR		
000090	5	1		ELIMI *	RESET AR	ZERO= RA	ADDWTE/852		
000100	6			ELIMI *	RESET R	ADDWTE*	ADDCU		

Help Specific to an Operation (ADDWT) and Parameters

- By placing your cursor next to any of the operations and using the Help key (F1), you can learn about the various parameters available for use (configuration) with this SAP-supplied operation

Documentation ADDWT - Add wage type to subsequent wage type (EN)

Link 

ADDWT - Add Wage Type to Subsequent Wage Type

Object

Operation

Use

The wage type affected is transferred to an internal table.

Input

Wage types that are in the header of the relevant input table (= work field) when processing is performed are the input values of this operation.

Processing

The values of the wage type in the current work field are added to the subsequent wage type. You also have the option of renaming the wage type or of storing it in a variable.

Output

In table determined by parameter S.

Syntax

```
00000SVVVV
```

00000	ADDWT	Name of operation
S		Result table of the operation
	blank	Output table OT
		Previous employee table WAG

Example Maintain/Display Operation – ADDWT

- Transaction PE04 is used to display, maintain, or create operations

Maintain Functions and Operations

Activate all Input parameters Output parameters

Name

Object class
 Payroll
 Time management

Object type
 Function
 Operation

Create Change Display Delete

Change



Example Maintain/Display Operation – ADDWT (cont.)

- Here you may display or maintain the characteristics of an operation, such as the country-specific use or parameters available for use
- You may create your own operations
- The standard ABAP name begins with OP

Characteristics Operation Payroll ADDWT

Operation: **ADDWT** ADDWT - Add wage type to subsequent wage type

Name of form routine:
 Standard name **OPADDW**
 Self-defined

Country assignment

Eligi...	Country Grouping
<input checked="" type="checkbox"/>	Argentina
<input checked="" type="checkbox"/>	Australia
<input checked="" type="checkbox"/>	Austria
<input checked="" type="checkbox"/>	Belgium
<input checked="" type="checkbox"/>	Brazil
<input checked="" type="checkbox"/>	Canada
<input checked="" type="checkbox"/>	China

Parameters

Model: **EB** Struct: **00000SV...** Own check routine

Parameter Vals								
Val. F	Val. S	Type V	Lngh V	Meaning	Check #	Reac	VkeyT	VkyLngh
		WGTY	4	OT Output table				
	&	ALL	4	VAR Variable table				
	A	WGTY	4	SICL				
	C	WGTY	4	CRT Cumulation tab.				
	D	WGTY	4	DT Difference tab.				
	E	WGTY	4	RT Results table				

When Are the Operations (ABAP) Executed?

- In this example, the executable schema Z000 (Config) will execute the subschemas ZT00 (Config), which will execute the function PIT (ABAP), which will execute the calculation rule Z020 – Store Gross Amount (Config), which will execute the Operation ADDWT (ABAP)

Payroll Driver (USA)

Selections from Search helps

Payroll period

Payroll area 12 09/19/2004

Current period

Other period 17 2004

Selection

Personnel number 10000099

Payroll area 12

General program control

Reason for payroll

Off-cycle payroll

Schema Z000

Forced retro.accounting as of

Test run (no update)

Log

Display log

Operations (ABAP) Determination and Verification

- The following payroll log information allows for the analysis of the payroll subschema ZT00, the function PIT, the calculation rule Z020, and the execution of the operations such as ADDWT

The screenshot displays the SAP payroll log interface. On the left, a tree view shows the hierarchy: PIT > Z020 P03 > CDC - Gross & RT storage time wage types > Processing > 1200 Regular working time. A red arrow points from this entry to the 'Detail View of Log' table on the right.

Detail View of Log

Rule	ESGPCR	VaKey	Operation
Z020	1		VWTCL 03
Z020	1	1	ADDWTE*
Z020	1	1	ELIMI KTX
Z020	1	1	ADDCU

Operations (ABAP) Determination and Verification (cont.)

- To determine and understand the ABAP code that is being executed for any specific operation, first you must set a break in the subschema prior to the calculation rule, execute payroll, and then set a breakpoint at the OPADDWT operation, as shown in the next slide



Operations (ABAP) Determination and Verification (cont.)

- Setting a breakpoint

000510	BREAK	CC				CDC - Tesing Break prior to Z020
000520	PIT	Z020	P03			CDC - Gross & RT storage time wage types

The screenshot shows the ABAP Debugger interface. The 'Breakpoints' menu is open, and the 'Subroutine...' option is selected. The 'Create Breakpoint' dialog box is displayed, showing the following fields:

- Program: RPCALCU0
- Subroutine/Method/Module: OPADDWT

The dialog box has a 'Breakpoint set' status box at the bottom right.



Operations (ABAP) Determination and Verification (cont.)

- “Run” and you arrive at the FORM OPADDWT ABAP code specific to the calculation rule Z020 (store gross amount)



Run (to Cursor) (F8)

Operations (ABAP) Determination and Verification (cont.)

- You can verify and analyze the processing of your configured use of the operation and the processing of your configured parameter for that operation via the standard debugger functionality



ABAP Debugger

Fields Table Breakpoints Watchpoints Calls Overview Settings

Main Program `RPCALCUB`

Source code of `RPCBU409_OPADDWT`

FORM OPADDWT

```
FORM opaddwt.                                "formerly ADDFL
  saveot = ot.
  PERFORM vcollect.
  ot = saveot.
ENDFORM.                                       "OPADDWT.
```

FORM VCOLLECT

```
IF ot-1gart NE space.
  CASE op-modif.
  WHEN ' '. COLLECT ot.
  WHEN 'E'.
    MOVE ot TO rt.
    COLLECT rt.
```

What We'll Cover ...

- HR/Payroll features (decision trees) – configuration
- Schemas and subschemas – configuration
- Payroll functions – SAP ABAP
- Calculation rules (or cycles) – configuration
- Operations within rules – SAP ABAP
 - ◆ Parameters used for operations – configuration
 - ◆ Parameters available for use – SAP ABAP
- Wage type integration – tie it together
- Wrap-up

The Concept of Wage Type Integration

- How functions are used in conjunction with wage type processing
- How the rules integrate with wage type processing
- How the operations and parameters to the operations can be varied by the use of the processing class attributes of the wage types
- Examine and understand the payroll log regarding the inputs, processing, and outputs of the wage type processing, and how each of the components described above is used to achieve the correct and customer-specific solution

Example Wage Type Integration – Processing Class 03

- In this example, this portion of the ZT00 subschema will execute the PIT (process input table) payroll function (ABAP) using a calculation rule of Z020 and a second parameter value of P03

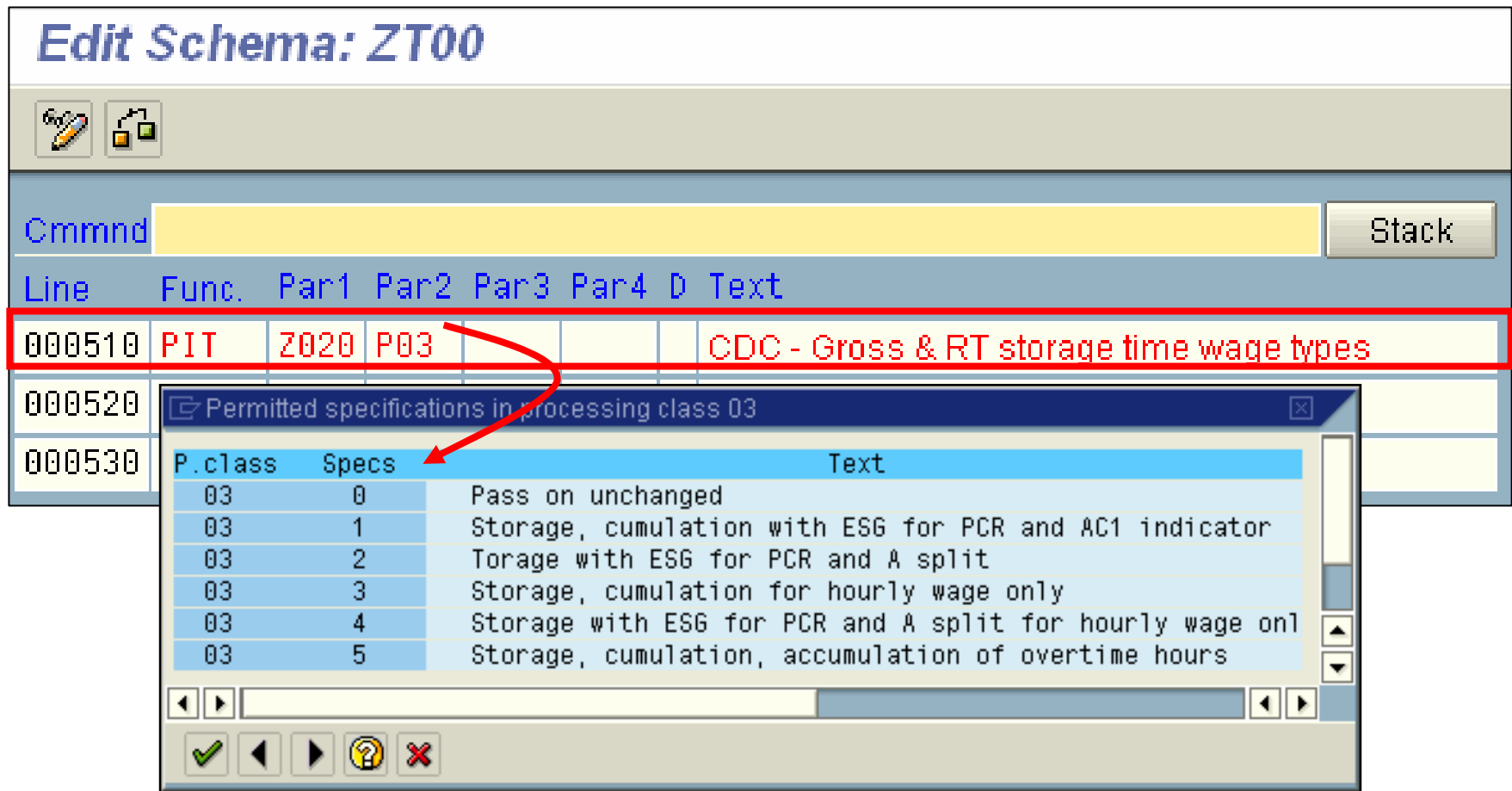
Edit Schema: ZT00

Cmmnd Stack

Line	Func.	Par1	Par2	Par3	Par4	D	Text
000510	PIT	Z020	P03				CDC - Gross & RT storage time wage types
000520							
000530							

Permitted specifications in processing class 03

P.class	Specs	Text
03	0	Pass on unchanged
03	1	Storage, cumulation with ESG for PCR and AC1 indicator
03	2	Storage with ESG for PCR and A split
03	3	Storage, cumulation for hourly wage only
03	4	Storage with ESG for PCR and A split for hourly wage onl
03	5	Storage, cumulation, accumulation of overtime hours



Example Wage Type Integration – Processing Class 03 (cont.)

- Wage types (earnings, tax, and deductions) have processing class values that may be configured. In my example, the processing class 03 value for wage type 1200 (regular working time) is 1

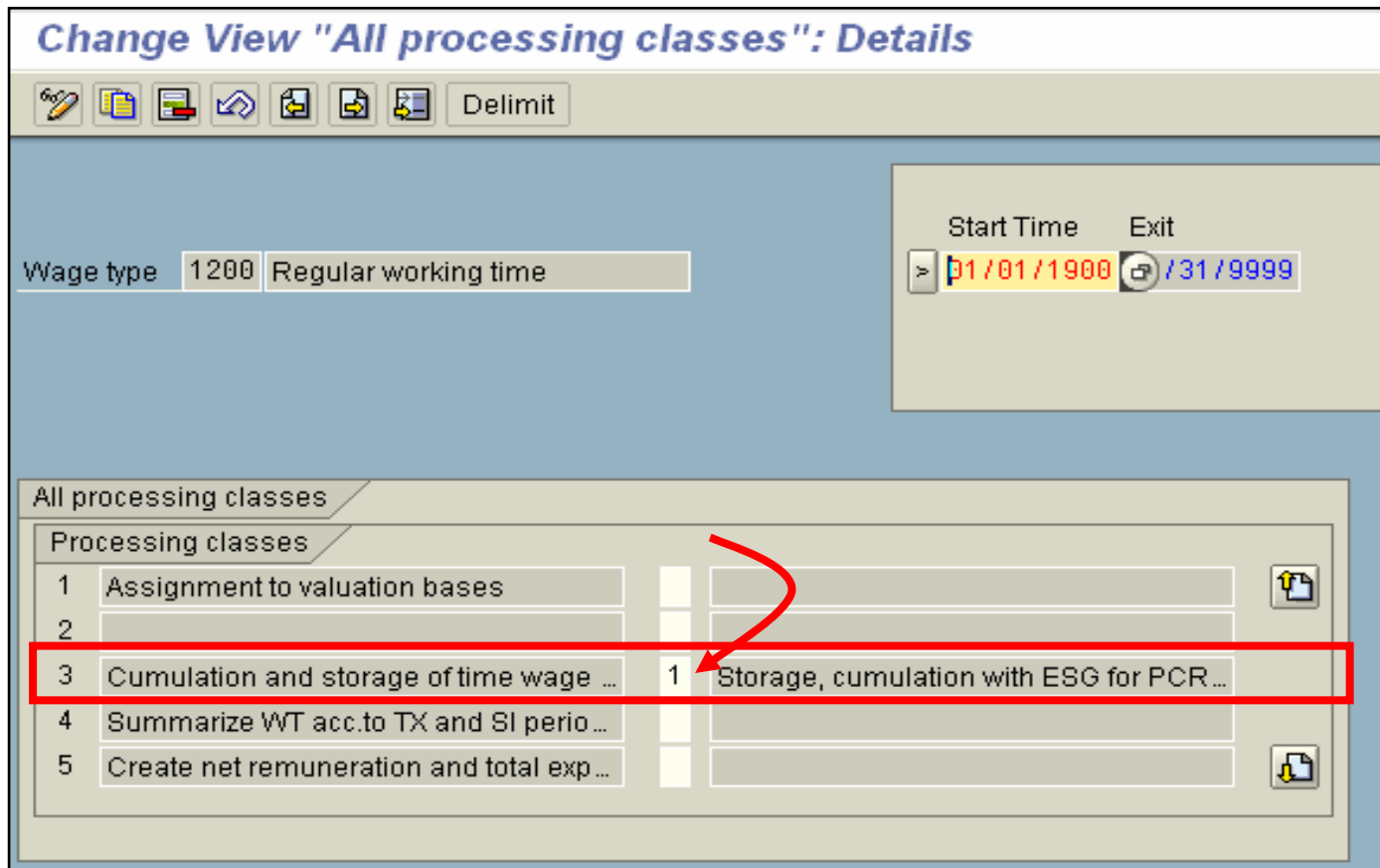
Change View "All processing classes": Details

Wage type 1200 Regular working time

Start Time 01/01/1900 Exit /31/9999

All processing classes



Processing classes	
1	Assignment to valuation bases
2	
3	Cumulation and storage of time wage ... Storage, cumulation with ESG for PCR ...
4	Summarize WT acc.to TX and SI perio ...
5	Create net remuneration and total exp ...



Example Wage Type Integration – Processing Class 03 = 1

- Here you can configure exactly which operations should be performed based on the wage types processing class 03 value

*Edit Rule: Z020 ES Grouping * Wage Type/Time Type *****

Cmmnd

Line	Var.Key	CL	T	Operation	Operation	Operation	Operation	Operation	Operation	*
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----										
000010				D	VWTCL 03					
000020	*				ERROR					
000030	0				ADDWT *					
000040	1				ADDWTE*	ELIMI KTX	ADDCU			
000050	2				ELIMI *	RESET AR	ADDWTE*			
000060	3				ADDWTE*	ELIMI KTX	ADDCU			
000070	4				ELIMI *	RESET AR	ADDWTE*			
000080	5				ADDWTE*	ELIMI KTX	ADDCU	NEXTR		
000090	5		1		ELIMI *	RESET AR	ZERO= RA	ADDWTE/852		
000100	6				ELIMI *	RESET R	ADDWTE*	ADDCU		

How Does it Fit Together (ABAP and Configuration)?

Payroll Driver (USA)

Selections from Search helps

Payroll period

Payroll area 12 09/19/

Current period

Other period 17 2004

Selection

Personnel number 10000099

Payroll area 12

General program control

Reason for payroll

Off-cycle payroll

Schema Z000

Forced retro.accounting as of

Test run (no update)

Log

Display log

- In this final example, the executable schema Z000 (Config) will execute the subschemas ZT00 (Config), which will execute the function PIT (ABAP), which will execute the calculation rule Z020 - Store Gross Amount (Config), which will examine the wage types processing class 03 value (Config), to determine which operations (ABAP) to execute, using the configured operation parameters

How Does It Fit Together (ABAP and Configuration)? (cont.)

- The following payroll log information allows for the analysis of the payroll subschema ZT00, the function PIT, the calculation rule Z020, and the processing of wage type 1200 with a processing class 03 value of 1

The screenshot displays the SAP payroll log interface. On the left, a tree view shows the processing steps under 'Processing'. A red arrow points from the '1200 Regular working time' entry in this list to the 'Detail View of Log' window on the right.

The 'Detail View of Log' window shows a table of log entries for rule Z020. The table has columns for Rule, ESGPCR, VaKey, and Operation. The 'VaKey' column contains the value '1' for all entries, which are highlighted with red boxes. Red arrows point from these '1' values to the 'VWTCL 03' operation in the first row, and from the '1' values in the subsequent three rows to the 'ADDWTE*', 'ELIMI KTX', and 'ADDCU' operations respectively. The 'VWTCL 03' operation is also enclosed in a red box.

Rule	ESGPCR	VaKey	Operation
Z020	1	1	VWTCL 03
Z020	1	1	ADDWTE*
Z020	1	1	ELIMI KTX
Z020	1	1	ADDCU

How Does It Fit Together (ABAP and Configuration)? (cont.)

- The end result is that all pieces fit together, allowing for the earnings wage types to be stored into the payroll results table and accumulated to allow for the continuation of the gross-to-net payroll accounting with /101 (total gross)

How Does It Fit Together (ABAP and Configuration)? (cont.)

- The end result (cont.)

PIT		Z020 P03		CDC - Gross & RT storage time wage types	
Input					
IT					
Processing					
Output					
IT					

1	1001	Hourly Rat 01				16.9600			
1	1200	Regular wo 01	X 01			16.9600	8.00		135.68
1	1200	Regular wo 01	X 02			16.9600	8.00		135.68
1	1200	Regular wo 01	X 03			16.9600	8.00		135.68
1	1206	Overtime p 01	W 01			25.67	7.50		192.53
1	1213	Night Shif	X 04			0.15	10.50		1.58
1	1213	Night Shif	X 05			0.15	10.50		1.58
1	1213	Night Shif	X 06			0.15	10.50		1.58
1	1251	Vacation p 01	X 07			16.9600	8.00		135.68
1	1251	Vacation p 01	X 08			16.9600	8.00		135.68

* /101 Total gros									875.67
-------------------	--	--	--	--	--	--	--	--	--------

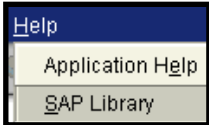
Finish



What We'll Cover ...

- HR/Payroll features (decision trees) – configuration
- Schemas and subschemas – configuration
- Payroll functions – SAP ABAP
- Calculation rules (or cycles) – configuration
- Operations within rules – SAP ABAP
 - ◆ Parameters used for operations – configuration
 - ◆ Parameters available for use – SAP ABAP
- Wage type integration – tie it together
- Wrap-up

Resources

- Online SAP Library  > SAP R/3 Enterprise Application Components > Human Resources > Payroll (PY) > Payroll United States (PY-US) >
 - Payroll in the SAP System > Payroll Basics (PY-XX-BS)
 - Gross and Net Part of Payroll (PY-US-GR & PY-US-NT)

- From any of the SAP selection screens:



- SAP Service Marketplace (www.service.sap.com) – United States > Online Documentation, Documentation, & Best Practices
- mySAP.com > Services > Tools, Tech & Methodologies > Best Practices > CRM > Building Block Library

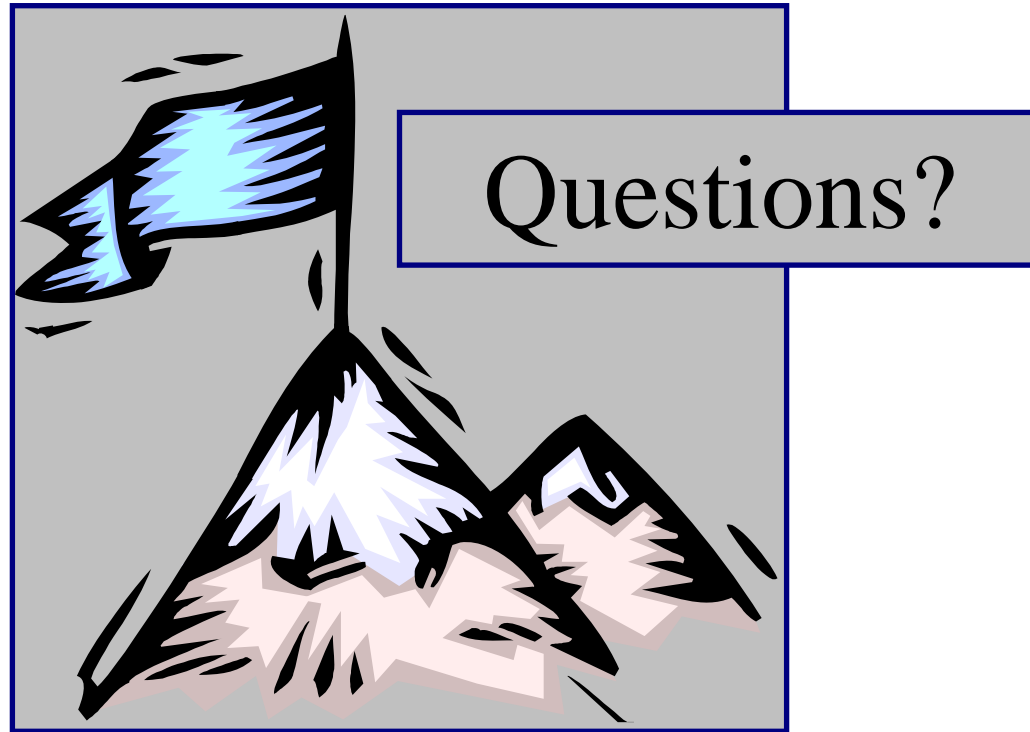
7 Key Points to Take Home

- HR/Payroll features are decision trees that typically execute at dialog entry time and are configured. They reduce data entry errors via logical default values
- Schemas and subschemas allow you to control the flow of processing and are configured to meet requirements. You pick and choose what to use
- Payroll functions are “objects” of SAP ABAP code that you have purchased and can choose to use or not to use. Custom functions can be created if necessary
- Calculation rules (or cycles) allow you the ultimate control of the processing within payroll and are configured to meet requirements

7 Key Points to Take Home (cont.)

- Operations are “objects” of SAP ABAP code that you have purchased and can choose to use or not to use within the calculation rules. The use of the operations in the rules are configured to meet requirements
- The parameters used for operations tie to SAP ABAP code that you have purchased and can choose to use or not to use for any specific operation. The use of the parameters for each operation are configured to meet requirements
- The integration of the wage type processing in payroll accounting is a primary example of how these “building blocks” tie together to achieve a complete, compliant, and business-specific solution for payroll accounting in SAP

Your Turn!



**How to Contact Me:
calc@rowsix.net**