Dirk Liepold and Steve Ritter

SAP® ERP HCM: Technical Principles and Programming
Contents at a Glance

1 Introduction to SAP ERP HCM .................................................. 25
2 Functional Overview for Developers ........................................ 35
3 Data Structures ........................................................................ 77
4 Reading and Processing HR Data ............................................. 147
5 Custom Enhancements ............................................................. 193
6 Enhancing Applications Using SAP Tools ................................ 221
7 HCM Reporting Tools .............................................................. 279
8 Developing Custom Reports .................................................... 343
9 HR Forms Workplace ............................................................... 417
10 HCM Interface Tools ............................................................. 469
11 HCM Data Conversion ............................................................ 515
12 Authorizations ........................................................................ 559
13 ESS and MSS Applications ..................................................... 605
14 HCM Performance Programming ............................................ 647
15 SuccessFactors Integration ..................................................... 673
A User Exits, BAdIs, and Enhancements ................................. 699
## Contents

Preface ....................................................................................................... 17  
Acknowledgments ..................................................................................... 23

### 1 Introduction to SAP ERP HCM ................................................ 25

1.1 Core Modules ............................................................................. 26  
1.1.1 Personnel Administration ............................................... 27  
1.1.2 Organizational Management .......................................... 27  
1.1.3 Payroll and Benefits Administration ................................. 28  
1.1.4 Time Management .......................................................... 29  
1.2 Other Modules ........................................................................... 30  
1.3 Summary .................................................................................... 32

### 2 Functional Overview for Developers ........................................ 35

2.1 Personnel Administration ............................................................ 35  
2.2 Organizational Management ....................................................... 40  
2.2.1 Objects, Infotypes, and Relationships ............................. 41  
2.2.2 OM Integration with Other Modules .............................. 43  
2.3 Time Management ...................................................................... 44  
2.3.1 Time Data Recording and Administration ....................... 44  
2.3.2 Cross-Application Time Sheet ......................................... 47  
2.3.3 Time Evaluation ............................................................. 49  
2.4 Payroll ........................................................................................ 52  
2.4.1 Payroll Infotypes ............................................................ 53  
2.4.2 Payroll Control Record ................................................... 57  
2.4.3 Payroll Calculation ......................................................... 59  
2.4.4 Post-Payroll Activities ..................................................... 60  
2.5 Talent Management .................................................................... 61  
2.5.1 SAP E-Recruiting ............................................................ 62  
2.5.2 Talent Management and Talent Development ................ 65  
2.5.3 Enterprise Compensation Management ......................... 67  
2.5.4 Performance Management ............................................. 70  
2.5.5 Enterprise Learning ........................................................ 72
# Reading and Processing HR Data

4.1 Logical Databases ........................................................................ 147
  4.1.1 PNP for Personnel Master Data ...................................... 152
  4.1.2 PNPCE for Personnel Master Data .............................. 156
  4.1.3 PCH for Personnel Planning ....................................... 158

4.2 Access without Logical Databases ................................................. 161
  4.2.1 Direct Table Access via SELECT Statement ...................... 161
  4.2.2 Macros ........................................................................... 164
  4.2.3 Function Modules .......................................................... 173
  4.2.4 BAPIs for Infotype Updates ............................................ 183
  4.2.5 Reading Cluster Tables ................................................... 184
  4.2.6 Features ......................................................................... 186

# Custom Enhancements

5.1 Customer Exits ............................................................................ 194
  5.1.1 Implementation ............................................................. 196
  5.1.2 Example: Prorating Absence Quota Accrual Based
    on Custom Proration Factor ........................................... 197

5.2 BAdIs .......................................................................................... 204
  5.2.1 Implementation ............................................................. 205
  5.2.2 Example: Automatic Approval for CATS_APPR_LITE ........ 209

5.3 Enhancement Points ................................................................... 212
  5.3.1 Implementation ............................................................. 213
  5.3.2 Example: Default Personnel Area from Infotype 1008
    in Infotype 0001 ............................................................ 215

5.4 Modifications ............................................................................. 216
  5.4.1 Implementing a Modification ......................................... 216
  5.4.2 Example: Infotype 0416 (Quota Compensation
    Modification) .................................................................. 217

5.5 Summary .................................................................................... 219

# Enhancing Applications Using SAP Tools

6.1 Personnel Administration ............................................................ 221
  6.1.1 PA Infotype Enhancement .............................................. 222
  6.1.2 PA Custom Infotypes ...................................................... 233
Contents

6.1.3 PA Infotype Processing in the DCIF ................................ 242
6.1.4 Features ......................................................................... 244
6.1.5 Dynamic Actions ............................................................ 245

6.2 Organizational Management ....................................................... 252
6.2.1 OM Infotype Enhancement ............................................ 253
6.2.2 Custom OM Infotypes .................................................... 255
6.2.3 Custom Objects .............................................................. 256
6.2.4 Custom Relationships ..................................................... 256

6.3 Time Management ...................................................................... 257
6.3.1 CATS .............................................................................. 257
6.3.2 Example: Validating CATS Absence/Attendance Entries ............................................................ 258
6.3.3 Enhancing Time Infotypes .............................................. 261

6.4 Payroll Calculation and Time Evaluation ...................................... 265

6.5 Summary .................................................................................... 277

7 HCM Reporting Tools ........................................................................ 279

7.1 Table Utilities ............................................................................. 280
7.1.1 Data Browser ....................................................................... 280
7.1.2 General Table Display ......................................................... 284

7.2 InfoSets ...................................................................................... 286
7.2.1 InfoSet Creation ................................................................. 286
7.2.2 InfoSet Assignment to User Groups ................................ 290
7.2.3 InfoSet Additional Fields ................................................ 292
7.2.4 InfoSet Switches ............................................................. 294

7.3 QuickViewer ............................................................................... 297
7.3.1 Query for InfoSet ................................................................ 298
7.3.2 Query for Table Join ....................................................... 301
7.3.3 Export Query to SAP Query ............................................ 302

7.4 InfoSet Query ............................................................................. 303
7.4.1 Create InfoSet Query ...................................................... 304
7.4.2 Start Report ................................................................... 307
7.4.3 Set Operations ............................................................... 308

7.5 SAP Query .................................................................................. 310
7.5.1 Create SAP Query ........................................................... 312
7.5.2 Multi-Line Lists .............................................................. 317
7.5.3 Report-to-Report Assignment ........................................ 319
7.5.4 SAP Query Programs .................................................... 320
<table>
<thead>
<tr>
<th>7.5.5</th>
<th>SAP Query Transport Tool</th>
<th>322</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.5.6</td>
<td>SAP Query Text Translations</td>
<td>323</td>
</tr>
<tr>
<td>7.6</td>
<td>Human Resources Information System</td>
<td>325</td>
</tr>
<tr>
<td>7.6.1</td>
<td>Create HIS Query</td>
<td>325</td>
</tr>
<tr>
<td>7.6.2</td>
<td>HIS Configuration</td>
<td>327</td>
</tr>
<tr>
<td>7.7</td>
<td>HCM Payroll Reporting</td>
<td>330</td>
</tr>
<tr>
<td>7.7.1</td>
<td>Payroll Infotypes</td>
<td>330</td>
</tr>
<tr>
<td>7.7.2</td>
<td>Wage Type Reporter</td>
<td>333</td>
</tr>
<tr>
<td>7.8</td>
<td>HCM System Audit Reporting</td>
<td>336</td>
</tr>
<tr>
<td>7.8.1</td>
<td>Log of Report Starts</td>
<td>336</td>
</tr>
<tr>
<td>7.8.2</td>
<td>Logged Changes in Infotype Data</td>
<td>338</td>
</tr>
<tr>
<td>7.9</td>
<td>Summary</td>
<td>340</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8</th>
<th>Developing Custom Reports</th>
<th>343</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1</td>
<td>Employee Master Data</td>
<td>343</td>
</tr>
<tr>
<td>8.1.1</td>
<td>Employee Selection</td>
<td>345</td>
</tr>
<tr>
<td>8.1.2</td>
<td>Employee Name</td>
<td>351</td>
</tr>
<tr>
<td>8.1.3</td>
<td>Repetitive Structures–DO VARYING</td>
<td>352</td>
</tr>
<tr>
<td>8.1.4</td>
<td>Date and Time Calculations</td>
<td>356</td>
</tr>
<tr>
<td>8.1.5</td>
<td>Basic Pay</td>
<td>356</td>
</tr>
<tr>
<td>8.1.6</td>
<td>Using Provide Statement or Macro with Time-Dependent Data</td>
<td>358</td>
</tr>
<tr>
<td>8.1.7</td>
<td>Reading Audit Log Cluster Data</td>
<td>364</td>
</tr>
<tr>
<td>8.2</td>
<td>Organizational Management Data</td>
<td>365</td>
</tr>
<tr>
<td>8.2.1</td>
<td>Reading OM Infotypes via Function Module</td>
<td>368</td>
</tr>
<tr>
<td>8.2.2</td>
<td>Commonly Used Macros for Logical Database PCH</td>
<td>369</td>
</tr>
<tr>
<td>8.3</td>
<td>Payroll Data</td>
<td>369</td>
</tr>
<tr>
<td>8.3.1</td>
<td>Reading Payroll Cluster Data</td>
<td>370</td>
</tr>
<tr>
<td>8.3.2</td>
<td>Payroll Results Infotypes</td>
<td>381</td>
</tr>
<tr>
<td>8.3.3</td>
<td>Custom Payroll Country Versions</td>
<td>383</td>
</tr>
<tr>
<td>8.4</td>
<td>Time Management Data</td>
<td>384</td>
</tr>
<tr>
<td>8.4.1</td>
<td>Time Management Infotypes</td>
<td>385</td>
</tr>
<tr>
<td>8.4.2</td>
<td>Reading Time Cluster Data</td>
<td>387</td>
</tr>
<tr>
<td>8.5</td>
<td>ABAP List Viewer Reporting</td>
<td>388</td>
</tr>
<tr>
<td>8.5.1</td>
<td>ALV Reporting Using Methods</td>
<td>389</td>
</tr>
<tr>
<td>8.5.2</td>
<td>ALV Reporting Using Function Modules</td>
<td>403</td>
</tr>
<tr>
<td>8.6</td>
<td>Summary</td>
<td>414</td>
</tr>
</tbody>
</table>
10.1.4 Local BAPI Execution .......................................................... 479
10.1.5 Remote BAPI Execution .................................................... 481
10.2 Interface Toolbox .................................................................. 482
  10.2.1 Interface Toolbox Configuration .................................. 484
  10.2.2 Interface Format Export ................................................ 489
  10.2.3 Export History .................................................................. 492
  10.2.4 File Layout Enhancement ................................................ 493
  10.2.5 Outsourcing Master Data & Payroll Results .................... 494
10.3 Application Link Enabling ....................................................... 499
  10.3.1 IDoc/ALE Messaging ....................................................... 499
  10.3.2 ALE Distribution ............................................................ 502
10.4 XML Data Interchange .......................................................... 504
  10.4.1 XML File Layout for Interface Formats ......................... 505
  10.4.2 XML Web Services ......................................................... 506
  10.4.3 XML Development ......................................................... 510
10.5 Summary .............................................................................. 514

11 HCM Data Conversion ............................................................ 515
11.1 Data Transfer Methods ........................................................ 515
  11.1.1 Direct Input ................................................................... 516
  11.1.2 Batch Input ................................................................... 517
11.2 Batch Data Communication .................................................. 523
  11.2.1 Creating BDC Sessions .................................................... 523
  11.2.2 Recording a Transaction ................................................ 526
  11.2.3 Processing BDC Sessions ................................................ 529
11.3 Legacy System Migration Workbench .................................... 533
  11.3.1 LSMW Recordings .......................................................... 534
  11.3.2 LSMW Objects ............................................................... 536
  11.3.3 LSMW Administration .................................................... 541
11.4 Techniques for HCM Data Conversion ................................... 543
  11.4.1 HCM Conversion Preparation ....................................... 544
  11.4.2 HCM Conversion Strategy .............................................. 549
  11.4.3 HCM Conversion Test Cycles ........................................ 555
11.5 Summary .............................................................................. 556
12 Authorizations

12.1 Authorizations Objects and Authorization Fields

12.1.1 Organization Key

12.1.2 Authorizations, Profiles, Roles, and Profile Generator

12.1.3 HCM Specific Authorization Objects

12.1.4 Double Verification Principle

12.1.5 Test Procedure

12.2 Structural Authorization

12.2.1 Defining and Assigning Structural Authorizations

12.2.2 Interaction of General and Structural Authorizations

12.2.3 Context-Based Authorization

12.2.4 HR Authorization Main Switches and Additional Functions

12.2.5 Periods of Responsibility and Time Logic

12.2.6 Common Issues

12.3 Authorization Enhancements

12.3.1 BAdIs

12.3.2 Customer Authorization Objects

12.3.3 Other Important Authorizations

12.4 Adding Authorization Checks to Custom Reports

12.5 Troubleshooting Authorization Issues

12.6 Summary

13 ESS and MSS Applications

13.1 HCM Self-Service Applications

13.1.1 Employee Self-Service

13.1.2 Manager Self-Service

13.1.3 HR Administrative Services

13.1.4 Employee Interaction Center

13.2 Enhancement Packages

13.2.1 Enhancement Package Components

13.2.2 Switch Framework

13.2.3 Activating a Business Function

13.3 SAP NetWeaver Business Client

13.3.1 SAP NetWeaver Business Client for Desktop

13.3.2 SAP NetWeaver Business Client for HTML
<table>
<thead>
<tr>
<th>13.4</th>
<th>Employee Self-Service (WDA)</th>
<th>618</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.4.1</td>
<td>ESS (WDA) Business Functions</td>
<td>619</td>
</tr>
<tr>
<td>13.4.2</td>
<td>Employee Navigation</td>
<td>620</td>
</tr>
<tr>
<td>13.4.3</td>
<td>Authorizations for ESS (WDA)</td>
<td>621</td>
</tr>
<tr>
<td>13.4.4</td>
<td>Customizing ESS (WDA) Applications</td>
<td>623</td>
</tr>
<tr>
<td>13.5</td>
<td>Manager Self-Service (WDA)</td>
<td>626</td>
</tr>
<tr>
<td>13.5.1</td>
<td>MSS (WDA) Business Function</td>
<td>627</td>
</tr>
<tr>
<td>13.5.2</td>
<td>Authorization for MSS (WDA)</td>
<td>628</td>
</tr>
<tr>
<td>13.5.3</td>
<td>Manager Navigation</td>
<td>629</td>
</tr>
<tr>
<td>13.5.4</td>
<td>Configuring MSS (WDA) Applications</td>
<td>634</td>
</tr>
<tr>
<td>13.5.5</td>
<td>Customizing MSS (WDA) Applications</td>
<td>638</td>
</tr>
<tr>
<td>13.6</td>
<td>Internet Communication Manager</td>
<td>641</td>
</tr>
<tr>
<td>13.6.1</td>
<td>Activating HTTP Services</td>
<td>641</td>
</tr>
<tr>
<td>13.6.2</td>
<td>Create/Change a Service</td>
<td>643</td>
</tr>
<tr>
<td>13.7</td>
<td>Summary</td>
<td>645</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>14</th>
<th>HCM Performance Programming</th>
<th>647</th>
</tr>
</thead>
<tbody>
<tr>
<td>14.1</td>
<td>HCM Performance Programs</td>
<td>647</td>
</tr>
<tr>
<td>14.1.1</td>
<td>Parallel Payroll Scheduler</td>
<td>648</td>
</tr>
<tr>
<td>14.1.2</td>
<td>Parallel Evaluation Scheduler</td>
<td>651</td>
</tr>
<tr>
<td>14.1.3</td>
<td>Parallel and Distributed Reporting Scheduler</td>
<td>652</td>
</tr>
<tr>
<td>14.1.4</td>
<td>Fast Entry for Master Data</td>
<td>655</td>
</tr>
<tr>
<td>14.2</td>
<td>Custom ABAP Development</td>
<td>660</td>
</tr>
<tr>
<td>14.2.1</td>
<td>Parallel Processing with Logical Database</td>
<td>660</td>
</tr>
<tr>
<td>14.2.2</td>
<td>Internal Table Processing</td>
<td>664</td>
</tr>
<tr>
<td>14.2.3</td>
<td>ABAP List Viewer</td>
<td>666</td>
</tr>
<tr>
<td>14.3</td>
<td>Summary</td>
<td>671</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>15</th>
<th>SuccessFactors Integration</th>
<th>673</th>
</tr>
</thead>
<tbody>
<tr>
<td>15.1</td>
<td>SAP ERP HCM and SuccessFactors Hybrid Models</td>
<td>675</td>
</tr>
<tr>
<td>15.2</td>
<td>Integration Add-On for SAP ERP HCM and SuccessFactors BizX</td>
<td>677</td>
</tr>
<tr>
<td>15.2.1</td>
<td>Integration Scenario for Employee Data</td>
<td>679</td>
</tr>
<tr>
<td>15.2.2</td>
<td>Adding Customer-Specific Fields</td>
<td>687</td>
</tr>
<tr>
<td>15.2.3</td>
<td>Integration Scenario for Compensation Data</td>
<td>689</td>
</tr>
<tr>
<td>15.2.4</td>
<td>Integration Scenario for Evaluation Data</td>
<td>693</td>
</tr>
<tr>
<td>15.2.5</td>
<td>Deletion of Employee Numbers and Extraction Results</td>
<td>695</td>
</tr>
</tbody>
</table>
The introduction to SAP standard enhancement technologies in the previous chapters has prepared you to explore more advanced SAP ERP HCM-specific techniques for enhancing on-premise SAP ERP HCM applications.

6 Enhancing Applications Using SAP Tools

HCM is probably one of the most complex yet most flexible applications in the SAP ERP suite. The infotype concept, dynamic actions, features, and the concept of time and payroll calculation rules using operations and functions are unique to HCM; they provide a wide variety of tools that allow you to enhance the standard-delivered processes and business process rules within SAP ERP HCM.

This chapter describes how you can adjust individual components of SAP ERP HCM using SAP-provided tools. We describe techniques for enhancing applications that involve more complex customization than would be part of a basic HCM system configuration. Most of the techniques explained in this chapter are unique to the HCM application. We focus on the core modules PA, OM, PT, and PY, and illustrate how to enhance existing infotypes to add custom fields, create new custom infotypes, and create custom objects and relationships. Besides infotype enhancements, we explain the use of features and dynamic actions and, at the end of this chapter, explain how to develop new customer-specific operations and functions for payroll and time evaluation.

6.1 Personnel Administration

To be able to enhance existing PA infotypes in the system, it is essential to understand the PA infotype concept, the technical components that make up an infotype, and the functionality and business logic that reside within the infotype module pool—as well as all infotype-related structures, configuration tables, program objects, BAdIs, customer exits, methods, and features that are included in a specific infotype.
Note

Before making the decision to enhance an existing standard SAP infotype, it is essential that you explore all other “less invasive” options (business process workaround, configuration, BAdI, customer exit, etc.) by involving functional PA experts. We have encountered many client scenarios where unnecessary enhancements have been applied due to the lack of knowledge of the functional PA area and available standard PA configuration options. Please refer to Chapter 2 for more details on the basic concept behind PA infotypes.

6.1.1 PA Infotype Enhancement

SAP offers the old enhancement option PBAS0001 using include ZXPADUnnn for extending standard infotype functionality and BAdI HRPAD00INFTY as a newer object-oriented (OO) approach. We illustrate both options in this chapter.

For enhancing infotype structures or building your own custom infotypes, use Transaction PM01 (see Figure 6.1) for PA infotype development; this transaction allows you to maintain and create all the individual components that make up a PA infotype. Recall that nnnn is four-digit unique infotype number. The following core components make up a PA infotype:

- **Structures:**
  - Pnnnn – Contains the key and header fields for the infotype; the structure serves as the interface between screens and dialog programs using this infotype
  - PSnnnn – Contains the data fields of the infotype

- **Module pool:** MPnnnn00

- **Screens:**
  - MPnnnn00 1000 (initial) (Note: do not change!)
  - MPnnnn00 2000 (standard)
  - MPnnnn00 3000 (list)

Some PA infotypes have country-specific screens. The last two digits of the screen number typically is equal to the two-digit country modifier (MOLGA).
Figure 6.1  Transaction PM01

Example: US-Specific Address Screen for Infotype 0006: MP000600 2010

In conjunction with infotype-specific features (Pnnnn), table T588M drives the screen flow and assigns alternative screens based on MOLGA and other attributes:

- Graphical user interface MPnnnn00 with necessary icons and function keys
- Dialog module RP_nnnn

We walk you through an example where, due to certain business requirements, additional fields have to be added to infotype 0006. Each infotype is defined by a table structure PSnnnn (revisit Chapter 2 for more details) and the majority of SAP standard infotypes contain an empty include structure CI_Pnnnn (as shown in Figure 6.2) within the infotype structure PSnnnn, which allows adding additional custom fields.

Some infotype structures do not have a CI_Pnnnn structure include and can therefore not be enhanced using this approach:

- 0000 – Actions
- 0302 – Additional actions
- 2nnn – Time Management infotypes (2001–Absences, 2002–Attendances, etc.)
Before you can add custom fields to an infotype structure, you have to create a domain in the SAP Data Dictionary (DDIC) that defines the field characteristics for your custom fields, unless you want to leverage existing domains from the SAP Data Dictionary to describe your new custom fields. The steps for creating a new domain are shown in Figure 6.3.
Once you have added your custom fields to structure CI_P0006 (return to Figure 6.2), saved, and activated the structure, you have to execute GENERATE OBJECTS from the main PM01 screen in order to apply the new fields to the module pool and infotype screens.

The system generates the following objects:

- **New screen** ZP000600 0200 with flow logic (you’ll see this in Figure 6.6 and Figure 6.7)
- **New module pool** ZP000600 (see Figure 6.4) with the following includes:
  - ZP000610 – Data definitions
  - ZP000620 – Process before output (PBO) routines
  - ZP000630 – Process after input (PAI) routines
  - ZP000640 – Form routines

You can display and further maintain the module pool and includes via Transaction SE38.

![ABAP Editor: Display Mod. Pool ZP000600](image)

**Figure 6.4** New Module Pool ZP000600

The new custom screen is automatically assigned to the main module pool via table T582C (Figure 6.5). You can display and further maintain the custom screen via Transaction SE51.
Transaction PM01 generates the required screen with all the screen attributes, the required flow logic (as shown in Figure 6.6), and the screen layout (as shown in Figure 6.7) automatically.
Note

For infotypes that do not have a CI_nnnn include in the infotype table structure PSnnnn, you can add additional custom fields only by modifying the SAP structure PSnnnn and manually adding the additional custom fields to an existing screen or copying an existing screen and adding the custom fields and custom flow logic for PBO and PAI processing. You can assign the custom screen via table T588M. You also have to ensure that the custom fields are manually added to structures Pnnnn and PAnnnn.

For creating infotype business logic enhancements to standard infotypes you can use SAP enhancement PBAS0001, which is shown in Figure 6.8. You have to create your own project via Transaction CMOD and then use one of the two function exits within this enhancement, which reside in the common infotype include MPPERS00. To add custom logic for process before output (PBO), use EXIT_SAPF50M_001 to add...
your custom code in include ZXPADU01; and for adding custom logic for process after input (PAI), use EXIT_SAPF50M_002 to add your custom code in include ZXPADU02.

Since the two exits are executed for all infotypes, you have to ensure that you use an IF or CASE statement to make your custom code infotype-specific, as shown in our code example in Figure 6.9, where we created different form routines for infotype 0007, 0008, and 0021 PBO custom code.

```
* Create new form routine for infotype 0007
 WHEN '0007'.
  IF i000p-molga EQ '10'.
   PERFORM pbo_0007_10 USING ipsyst 1503
   CHANGING innmn.
  ENDIF.
  WHEN '0008'.
  IF i000p-molga EQ '04'.
   PERFORM pbo_0008_04 USING ipsyst 1503
   CHANGING innmn.
  ENDIF.
  WHEN '0021'.
   PERFORM pbo_0021 USING ipsyst.
 END_CASE.
```
The example code shown in Figure 6.10 illustrates the code in form routine pbo_0008_04, which is called for infotype 0008 for Spanish employees only (MOLGA = 04) to manipulate the display of the annual salary field P0008-ANSAL during an infotype creation (IPSYST-IOPER = INS) for certain employee subgroups (T503-PERSK). The form routines are defined in include ZXPADF04.

```
ABAP Editor: Change Include ZXPADF04

Include ZXPADF04

FORM pbo_0008_04 USING ipsysyst TYPE ipsysyst
i503  TYPE t503

CHANGING innnn TYPE preip.

DATA: l_p0008  TYPE p0008,
     l_bet01  TYPE p0008-bet01,
     l_bet02  TYPE p0008-bet02.

CHECK ipsysyst-ioper = 'INS'.

CHECK innnn-begbe NE space.

CALL METHOD cl_hr_panna_type_cost->preip_to_pannn
  EXPORTING
    preip = innnn
  IMPORTING
    pannn = l_p0008.

IF i503-persk = '03' OR "PT-AUTO-UZ K1g"
  i503-persk = '06' OR "PT-AUTO-UZ K1g"
  i503-persk = '26' OR "PT-DAZZY ZS"
  i503-persk = '27' OR "PT-DAZZY ZS"
PERFORM get_period_cost_pbo USING l_p0008 i503
  CHANGING l_bet01 l_bet02.

CHECK l_bet01 EQ 0.

IF i503-persk = '26' OR "PT-DAZZY ZS"
  i503-persk = '27', "PT-DAZZY ZS"
  l_p0008-ansal = ( l_bet01 + l_bet02 ) * 30.
ELSE.
  l_p0008-ansal = ( l_bet01 + l_bet02 ) * 43.
ENDIF.

CALL METHOD cl_hr_panna_type_cost->pnnan_to_preip
  EXPORTING
    pnnan = l_p0008
  IMPORTING
    preip = innnn.
ENDIF.

ENDFORM.
```

Figure 6.10  Include ZXPADF04 with Form pbo_0008_04
Figure 6.11 shows an example for include ZXPADU02, which is used in exit EXIT_SAPF50M_002 for PAI processing. In the example, the permissibility of wage type ‘9999’ in field P0008-LGA01 for employees with employee subgroup ‘01’ in field IPSYST-PERSK is checked and a message created in case wage type ‘9999’ is entered for an employee in employee subgroup ‘01’.

Structures PSYST (see Figure 6.12), I503, and I001P contain important system and employee information that you can use in your custom logic. Structure PREL is used to import the current infotype data and export any changed infotype data after changing in your custom logic.

The older version of the custom exit is executed first—after that, the new BAdI enhancement is called. This is true for both PBO and PAI processing. Consider the example for the PBO custom hooks in include MPERS00 in Figure 6.13.

To use BAdI HRPAD00INFTYPE, you have to create a BAdI implementation and add your custom code in the appropriate method within this BAdI. See example ZHR_EU_INFTY in Figure 6.14.
Figure 6.12  Structure PSYST

Figure 6.13  PBO Custom Hooks in Include MPPERS00
The structures you can use for your custom logic are similar to the structures for the customer exit. The code example in Figure 6.15 shows method
with custom code to check the person ID (P0002-\text{PERID}) in infotype 0002 for Spanish employees (MOLGA = 04) and create an error message if it is ‘99999999R’.

### 6.1.2 PA Custom Infotypes

In some cases it is not sufficient to add data fields to an existing infotype or enhance the business logic of an existing infotype, so you are required to build your own custom infotype in order to store customer-specific data. A common example of the need to create a custom infotype is the requirement to store historical HR data that cannot be saved in standard SAP infotypes, but that need to be available for end users to display and maintain per employee from within SAP ERP HCM.

In our example, the client requested an infotype to store specific union information per employee that could not be stored in any existing SAP infotype. These steps outline the process for developing custom infotype 9002 for this purpose:

1. Define type of infotype (international or country-specific version).
2. Define infotype structure.
3. Define enhancement category.
4. Generate infotype objects.
5. Maintain infotype attributes.
6. Test infotype.
7. (As necessary) Apply additional processing logic.
8. (As necessary) Adjust infotype screen.

**Note**

The customer name range for custom PA infotypes is 9000 to 9999.

The first step is to define custom infotype structure PS9002, as shown in Figure 6.16. Here you define all the fields that should be saved in the infotype. It is possible to reference existing SAP component types or to define your own custom component types.
You can define custom check tables and link a component entry to this check table to ensure automatic data validation against entries within this check table during infotype maintenance via PA30. To check for valid union codes, table ZHR_UNION_CODES (see Figure 6.17) was created via Transaction SE11 to allow maintaining valid union codes via Transaction SM31 (Table Maintenance).
You can link component `UNION_CD` of structure `PS9002` to the check table `ZHR_UNION_CODES` by going to the `ENTRY HELP/CHECK` tab and assigning the check table as shown in Figure 6.18.

After defining, saving, and activating the new infotype structure `PS9002`, we now have to define the infotype characteristics for our new infotype `9002`. The infotype characteristics determine the overall behavior of the infotype maintenance functionality.

Infotypes and their characteristics are defined for each infotype in table `T582A`. You can access table maintenance from within Transaction `PM01` via `INFOTYPE CHARACTERISTICS` or directly from Transaction `SM31`, as shown in Figure 6.19.
Let's take a look at the most important attributes shown in Figure 6.19 and their impacts:

- **TIME CONSTRAINT**: A time constraint indicates whether more than one infotype record may be available at one time. The following are possible time constraint indicators:
  - 1: An infotype record must be available at all times. This record may have no time gaps. You may not delete the record last stored on the database because all records of this infotype would be deleted (examples: infotypes 0001, 0002, and 0007).
  - 2: Only one record may be available at one time, but time gaps are permitted (examples: infotypes 0041 and 0019).
  - 3: Any number of records may be valid at one time, and time gaps are permitted (examples: infotypes 0037 and 0040).
  - A: Only one record may ever exist for this infotype from 01/01/1800 to 12/31/9999. Splitting is not permissible. Infotypes with time constraint A cannot be deleted (examples: infotypes 0003 and 4004).
B: Only one record may ever exist for this infotype from 01/01/1800 to 12/31/9999. Splitting is not permissible. Infotypes with time constraint B can be deleted (examples: infotypes 0031 and 0130).

T: The time constraint varies depending on the subtype. The subtypes are defined in a separate table, table T591A (examples: infotypes 0006 and 0021).

Z: Used for time management infotypes. The time constraint for these infotypes depends on the time constraint class defined in view V_T554S_I (Absence: General Control). Collision checks are defined in view V_T554Y (Time Constraint Reaction) (examples: infotypes 2001, 2002, and 2006).

Time constraint table: The time constraint for some infotypes is defined in more detail via a separate time constraint table.

Subtype obligatory: A subtype is required for this infotype.

Subtype text table: This attribute is a customization table where the subtype texts are defined.

Subtype field: This is where you define which field defines the subtype.

Text allowed: If you check this attribute, texts can be entered for each infotype record via Maintain Text or F9 from Transaction PA30. The texts are stored in table PCL1 cluster TX and the infotype header field ITXEX is set to ‘X’ when a text exists for an infotype record.

Indicator for entry of additional data: This attribute defines whether or not you can enter additional controlling objects for the infotype.

Access authorization: The access authorization parameter allows you to define the time period during which an infotype can be accessed. If you do not set this indicator (initial value), the administrator is authorized to access the infotypes if the person had, has, or will have an organizational assignment, which, in accordance with the authorization profile, allows him/her to access this data. If you set this indicator (X), the authorization check depends on the current (system) date.

Display and reaction: The three date reaction indicators control what type of databank search is carried out when no dates, one date, or two dates are entered on the selection screen.

Sort sequence: This attribute defines the sort sequence of records.

Screen header: This attribute allows assigning different infotype header formats for the screen display. Header IDs are assigned a header modifier, from which...
the structure of the respective headers is defined, as well as the HR country modifier and transaction class in table T588I (Assignment of Header ID/Country Grouping to Header Modifier).

- **Retroactive Accounting Payroll:** If this attribute is set, infotype changes will cause a payroll retro-calculation (infotype 0003 [Earliest Change in Master Data Since Last Payroll] will be set upon changes to the infotype).

- **Retroactive Accounting PDC:** If this attribute is set, infotype changes will cause a time evaluation retro-calculation (infotype 0003 [Retroactive accounting date for PDC] will be set upon changes to the infotype).

The following are possible retro flag values:
- ‘ ’ = Infotype is not relevant to retroactive accounting
- ‘R’ = Change triggers retroactive accounting
- ‘T’ = Retroactive accounting is field-dependent according to T588G

- **Past Entry Allowed:** This attribute defines the permissibility of infotype data maintenance for payroll past.

- **Single Screen:** This attribute defines the default maintenance screen. This screen can be dynamically adjusted using table T588M for dynamic screen adjustment. The name range for customer screens is 2900 to 2999.

- **List Screen:** This attribute defines the default list screen. This screen can be dynamically adjusted using table T588M for dynamic screen adjustment. The name range for customer screens is 3900 to 3999.

- **Dialog Module:** RP_nnnn

- **Structure:** Pnnnn

- **Database Table:** PAnnnn where nnnn is the custom infotype number

After defining the infotype characteristics for our new infotype, we now have to define the technical attributes of the infotype. Infotypes and their technical attributes are defined for each infotype in table T7770. You can access table maintenance from within Transaction PM01 via TECHNICAL ATTRIBUTES or directly from Transaction SM31 (Table Maintenance), as shown in Figure 6.20.

Each custom infotype needs to at least have the following entries in table T7770, which are already defaulted from Transaction PM01:
After all required infotype components for our custom infotype 9002 have been created, you can generate the module pool and infotype screens for the new infotype by clicking the GENERATE button from within Transaction PM01. This will generate the basic module pool MP900200 as shown in Figure 6.21, Figure 6.22, and Figure 6.23, and all the required includes (MP900210, MP900220, MP900230, MP900240, MP900250) plus the basic screens MP900200 1000, 2000, and 3000, including the flow logic. It will also create the corresponding P9002 structure, create table PA9002, generate entries in tables T777D and T771D, and, finally, generate the screen structure ZHCMT_BSP_US_R9002. This screen structure contains the same...
fields as structure PS9002 and check class ZCL_HRPA_INFOTYPE_9002, plus an entry in tables T582UICONVCLAS and T582ITVCLAS.

As a rule of thumb, the infotype-specific check class ZCL_HRPA_INFOTYPE_9002 inherits the properties of the super class CL_HRPA_INFOTYPE_nnnn. Also, any country-specific check class inherits the properties of the infotype check class and the super class.

**Figure 6.21** Automatically Generated Module Pool MP900200

```abap
* Generated by 039999 02.01.2012 12:39:13
1 INCLUDE EP900210. " Data Definitions
2 INCLUDE EP900210. " Global Data - Do not change
3 INCLUDE PS9002. " Local Data - Do not change
4 INCLUDE EP9002. " Standard Modules- Do not change
5 INCLUDE PS9002. " Input Modules
6 INCLUDE PS9002. " Subroutines
```

**Figure 6.22** Automatically Generated Infotype Screen Flow Logic MP900200

```abap
1 PROCEDURE BEFORE OUTPUT.
2 " general infotype-independent operations
3 MODULE BEFORE_OUTPUT.
4 CALL SCREEN subscreen_empl INCLUDING empl_prog empl_dyna.
5 CALL SCREEN subscreen_header INCLUDING header_prog header_dyna
6 " infotype specific operations
7 MODULE p9002.
8 " MODULE hidden_data.
9 " PROCEDURE AFTER INPUT.
10 " processing after input
11 " check and mark if there was any input: all fields that
12 " accept input HAVE TO BE listed here
13 " CHAIN.
14 FIELD p9002-begda.
15 FIELD p9002-endda.
16 FIELD p9002-comment33.
17 FIELD p9002-comment34.
18 FIELD p9002-comment31.
19 FIELD p9002-end_of_prob.
20 FIELD p9002-union_steward.
21 FIELD p9002-union_id.
```
If you want to apply additional code changes to the automatically generated module pool code or to the includes, you can access all objects individually via Transactions SE80 or SE38. The infotype screens can be changed via Transaction SE51 after the automatic generation. Infotype tables and structures can also be changed after the automatic generation of the new infotype via Transaction SE11. You have to be careful making changes after your module pool; screens and user interfaces have been generated via Transaction PM01 in order to avoid inconsistencies in your custom infotype. You can get a good overview of all objects (see Figure 6.24) related to your custom infotype by using the Repository Browser in Transaction SE80 and selecting program MP900200.
6.1.3 PA Infotype Processing in the DCIF

Most of the standard and custom infotypes have been developed using the traditional way of defining all of the business logic in the infotype module pool, including the plausibility checks, defaulting and displaying of data on the infotype screen and infotype table read, and update processing. The core concept of process before output (PBO) and process after input (PAI)—which originated from the R/2 assembler mainframe programs, macros, and dynpros and was then adapted to the R/3 client-server technology into ABAP—is still very much present in most of the current infotypes. The new technology for infotype processing is based on decoupled business logic from the transactional display (view). In the DCIF the following three entities must exist:
Screen structure (which is a DDIC structure defining the infotype fields)
ABAP-OO class (also known as conversion class), which takes care of converting the data from the back-end to the front-end UI
Entry in table T588UICONVCLAS to assign conversion class to screen structure

Instead of using the module pools containing all the business logic, the business logic now resides in so-called check classes. For each possible master data access scenario, SAP provides a method defined by the interface IF_HRPA_INFOTY_BL. SAP delivers a default implementation class CL_HRPA_INFOTYPE_nnnn.

Besides being able to implement business logic for infotypes via object-oriented methods, SAP also provides a new way to access infotypes via classes. We mentioned the DCIF concept in the context of infotype development because SAP is slowly converting the module pool-based code to object-oriented (OO)-based code following the DCIF concept. SAP also recommends using the DCIF for any new custom infotype development.

So what does that mean for you during custom development?

You need to know whether an infotype or a version of an infotype is DCIF-enabled (check view V_T582ITVCLAS).

If an infotype supports both the module pool and DCIF, then you need to know whether the PC_UI master switch is set. If it is not set, then infotypes accessed via PA30 would only process the module pool code.

Most processes and forms and all XSS services access infotypes only via DCIF.

If you have implemented BAdIs and enhancement spots for the old—that is, not decoupled—and want an infotype to act in the same way in applications that use DCIF (such as Employee Self-Services and HR Administrative Services), you must also implement BAdI or enhancement spot HRPAD00INFTYDB. You can use this BAdI to implement enhancements for performing actions after updating the data from the infotype framework on the database.

The IN_UPDATE method of the old HRPAD00INFTY BAdI corresponds to the UPDATE_DB method of the new HRPAD00INFTYDB BAdI.

SAP also offers enhancements spot HRPAD00INFTYUI using the BAdI with the same name to allow modification of the user interface for infotypes using the DCIF. There is no counterpart of this option for non-decoupled infotypes because it is based on the screen structure concept.
6.1.4 Features

In many cases, it is preferable to use an SAP feature rather than a mapping table to map one set of values to another. SAP has many standard features, but it is possible to create custom features using Transaction PE03, as we have described in Chapter 4.

You can call a feature from within infotype processing by calling the routine `get_default` in include `MPnnnnn40`.

To complete this task, you can use the fields of structure `PSYST` (System Fields for HR Infotypes) as a decision structure; this structure contains a list of useful parameter values that you can access during infotype processing, as shown in Figure 6.25.

![Figure 6.25](image-url)
It is more common with features to use structures that start with “PME.” When fields of feature structures are to be used for the decision structure, they should be made available accordingly in advance. You can call the feature by using the function module `HRFEATURE BACKFIELD`. Even though the structure can be changed to include fields on which decisions can be made, the ABAP code that calls the feature at run time may not have the necessary value at that point; therefore, the decision is never reached.

Listing 6.1 offers a code example of a feature call.

```abap
Data: structure like pme04.
Data: feature like t549b-namen.
Data: back like t549a-abkrs.
Data: status(1).
*
  structure-molga = '10'.
  structure-persk = '01'.
  feature = 'ABKRS'.
*
  call function 'HR_FEATURE_BACKFIELD'
    exporting
      feature = feature
      struc_content = structure
      kind_of_error = space
    importing
      back      = back
    changing
      status    = status
  exceptions
    dummy            = 1
    error_operation       = 2
    no_backvalue        = 3
    feature_not_generated    = 4
    invalid_sign_in_funid    = 5
    field_in_report_tab_in_pe03 = 6
    others           = 7.
```

Listing 6.1 Example Feature Call

### 6.1.5 Dynamic Actions

Dynamic actions (called dynamic events in earlier SAP releases) are activities triggered automatically during infotype maintenance. Dynamic actions are unique to the SAP
ERP HCM application; they are a powerful tool that can be used to simplify HCM data processing without having to develop any custom code or modifying SAP standard code. The configuration for dynamic actions is stored in configuration table T588Z. Each infotype module pool MPnnnn00 includes the standard program MPPERS00. This program contains routines for calling the dynamic action functionality that is defined in table T588Z. During each PAI processing of the infotype screen, existing dynamic actions relevant to the respective infotype and function used are executed.

You can maintain dynamic actions in table T588Z either from within the IMG path PERSONNEL MANAGEMENT • GLOBAL SETTINGS IN PERSONNEL MANAGEMENT • BASIC SETTINGS • INFOTYPES • CREATE DYNAMIC ACTIONS or directly via Transaction SM31.

Our example in Figure 6.26 shows a dynamic action that is executed during the hiring action (MASSN EQ Z0 or ZN) for US employees (MOLGA EQ 10). The dynamic action will create an infotype 0041 record defaulting four date entries with default dates based on the begin date of infotype 0000.

<table>
<thead>
<tr>
<th>New Entries</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Change View “Dynamic Actions”: Overview</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td><strong>Field N</strong></td>
</tr>
<tr>
<td>0000</td>
<td>04 10</td>
</tr>
<tr>
<td>0000</td>
<td>04 11</td>
</tr>
<tr>
<td>0000</td>
<td>04 12</td>
</tr>
<tr>
<td>0000</td>
<td>04 13</td>
</tr>
<tr>
<td>0000</td>
<td>04 15</td>
</tr>
<tr>
<td>0000</td>
<td>04 16</td>
</tr>
<tr>
<td>0000</td>
<td>04 17</td>
</tr>
<tr>
<td>0000</td>
<td>04 18</td>
</tr>
<tr>
<td>0000</td>
<td>04 19</td>
</tr>
</tbody>
</table>

Figure 6.26 Table Maintenance View V_T588Z selecting Work Area ‘0000’

Dynamic actions consist of the following components:

- **INFOTYPE NUMBER (INFTY):** Specifies the infotype for which you want the dynamic action triggered (in T588Z maintenance, this is also called work area)
- **SUBTYPE (SUBTY):** Narrows the focus to a specific subtype
- **FIELD NAME (FIELDN):** Starts your action when a maintenance function is performed on a particular field
- **FUNCTION (FC):** Specifies the various maintenance operations to which your dynamic action may respond:
02 – Change
04 – Create
06 – Create and change
08 – Delete
10 – Change and delete
12 – Create and delete

SEQUENCE NUMBER (NO): Refers to a sequential number

STEP (A): Specifies a particular type of action; no dynamic action is executed if the function character has a value other than one of the following:

- P – Plausibility checks, which are used for checking conditions
- I – Calls an infotype for processing using one of the following actions:
  - INS – Create
  - MOD – Change
  - COP – Copy
  - LIS9 – Delimit
  - DEL – Delete

**Note**
The following syntax has to be used: action, infotype, subtype, object ID, from date, to date.

- W – Called after the ‘I’ statement and is used to set default values to screen fields while creating or copying another infotype record through the ‘I’ statement
- F – Calls a form routine (subroutines in an ABAP program) during your action
- V – Allows combining a number of fields for which you want to define a common dynamic action
- M – Sends SAP Office mail

**VARIABLE FUNCTION PART:** Defines the action to be performed and specifies the processing details when the dynamic action is triggered
Now that we have described the components and syntax for dynamic actions, we move on to explore real-life examples of dynamic actions to provide you with references for future development of dynamic actions. This list illustrates the flexibility and power of dynamic actions. If used intelligently, the use of dynamic actions can help avoid more complex enhancements in many cases.

**Example 1**

When a user creates or changes a field on infotype 0001, a dynamic action should be started.

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td></td>
<td></td>
<td>06</td>
<td>01</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Here we want the dynamic action to be processed for creates and changes to infotype 0001, so we use value “06” in the FO field. The FO column tells the dynamic action to process when infotype 0001 is changed or created.

Notice that the Field column is blank. Since we need the dynamic action to process if a user changes *any* fields on infotype 0001, we leave this field blank. This tells SAP to check all fields for change or create actions on this infotype.

**Example 2**

When a user changes the Employee Group (PERSG) field on infotype 0001, we want to start a dynamic action.

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td></td>
<td>PERSG</td>
<td>02</td>
<td>01</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Here, we limit our dynamic action processing to changes to the employee group field only (FO” = “02) and (FIELD = “PERSG”) for infotype 0001 (ITYPE = “0001”).

**Example 3**

When a user creates or changes any field on infotype 0001, we want to process the dynamic action if the employee’s company code is “1000.”
The dynamic action will only be processed further in the event that the company code for the relevant employee is “1000.”

**Example 4**

When a user creates or changes any field on infotype 0001, we want to process the dynamic action if the employee's company code is “1000” and the employee group is “1.”

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td></td>
<td></td>
<td>06</td>
<td>01</td>
<td>P</td>
<td>P0001-BUKRS = '1000'</td>
</tr>
</tbody>
</table>

This dynamic action will be processed only if the relevant employee's company code is “1000” and the employee group is “1.” It is important to note that two consecutive “P” statements without other operators imply a logical **AND**. Both statements must be true for the event to be processed further.

**Example 5**

When a user creates or changes any field on infotype 0001, we want to process the dynamic action if the employees' company code is “1000” or the employee group is “1.”

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td></td>
<td></td>
<td>06</td>
<td>02</td>
<td>P</td>
<td>P0001-PERSG = '1'</td>
</tr>
</tbody>
</table>

This example differs from Example 4 in that the operator “/X” is used. This character has the effect of changing the normally implied **AND** to an **OR**. In this example, the event will continue processing if the relevant employee's company code is “1000” or the employee group is “1.”
**Example 6**

When a user saves an infotype 0002 for a married employee, we want to bring up infotype 0021 to allow the user to enter the spouse dependent.

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0002</td>
<td></td>
<td></td>
<td>06</td>
<td>01</td>
<td>P</td>
<td>P0002-FAMST = '1'</td>
</tr>
<tr>
<td>0002</td>
<td></td>
<td></td>
<td>06</td>
<td>02</td>
<td>I</td>
<td>INS, 0021,1,,(P0002-BEGDA), (P0002-ENDDA)</td>
</tr>
</tbody>
</table>

If the marital status on infotype 0002 is “1”, the system interrupts the normal processing of the personnel event and creates (“INS”) an infotype 0021 (subtype 1) that has beginning and end dates equivalent to that of the employee’s infotype 0002. For entries that weren’t relevant for the dynamic action (object ID in this case), a comma fills that spot. The object ID portion of the syntax is rarely used.

**Example 7**

When a user saves an infotype 0002 for a married employee (FAMST = ‘1’), we want to bring up infotype 0021 to allow the user to enter the spouse dependent.

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0002</td>
<td></td>
<td></td>
<td>06</td>
<td>01</td>
<td>P</td>
<td>P0002-FAMST = '1'</td>
</tr>
<tr>
<td>0002</td>
<td></td>
<td></td>
<td>06</td>
<td>02</td>
<td>I</td>
<td>INS, 0021,1,,(P0002-BEGDA), (P0002-ENDDA) /D</td>
</tr>
</tbody>
</table>

In this example, otherwise identical to Example 6, the /D indicator is used to create the infotype 0021 in the background.

**Example 8**

When an employee has an employee subgroup containing a “U”, we want to create an infotype 0057 with a default wage type of M999.

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>PERSK</td>
<td>06</td>
<td>01</td>
<td>P</td>
<td>P0001-PERSK CA ‘U’</td>
<td></td>
</tr>
</tbody>
</table>
When the employee subgroup is changed during a create or change of infotype 0001, the event first checks the employee subgroup to see whether it contains a “U.” If it does, it then creates an infotype 0057 with the same beginning and end dates as infotype 0000. Since no subtype is specified, the user is prompted by the system to pick one. When the infotype appears, the wage type will be automatically set to M999.

**Example 9**

We want infotype 0041 to be copied any time the personnel area or employee group fields are affected while creating or changing infotype 001. To do this, we set the following conditions:

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td></td>
<td>WERKS</td>
<td>06</td>
<td>01</td>
<td>V</td>
<td>PERSG</td>
</tr>
<tr>
<td>0001</td>
<td></td>
<td>WERKS</td>
<td>06</td>
<td>02</td>
<td>I</td>
<td>COP, 0041,..,(P0000-BEGDA), (P0000-ENDDA)</td>
</tr>
</tbody>
</table>

This dynamic action groups personnel area (WERKS) with employee group (PERSG). When either field is affected during a create or change of infotype 0001, the dynamic action will copy infotype 0041, giving it the same begin and end dates as infotype 0000.

**Example 10**

When infotype 0006, subtype 1 is created or changed and the zip code field (PSTLZ) is affected, we want to execute the form routine CREATE_IT207 within custom program ZPX0001.

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0006</td>
<td>1</td>
<td>PSTLZ</td>
<td>06</td>
<td>01</td>
<td>F</td>
<td>CREATE_IT207(ZPX00001)</td>
</tr>
</tbody>
</table>
In the event of the zip code \(\text{PSTLZ}\) changing on infotype 0006 subtype 1, subroutine \text{CREATE\_IT207}\ within program \text{ZPX00001}\ is called. Once the external routine is started, it will process until its conclusion before going back to the normal personnel event and return to standard module pool processing. If the form \text{CREATE\_IT207}\ had been in the module pool (program) that was running at the time the event was executed, the program name could have been omitted.

**Example 11**

We want to define an email to be sent every time an employee’s position is affected as a result of a create or change to infotype 0001.

<table>
<thead>
<tr>
<th>Itype</th>
<th>Sty</th>
<th>Field</th>
<th>FO</th>
<th>No</th>
<th>A</th>
<th>Variable Function Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td></td>
<td>PLANS</td>
<td>06</td>
<td>01</td>
<td>M</td>
<td>M0001</td>
</tr>
</tbody>
</table>

This dynamic action sends an email every time an employee’s position is created or changed. The characteristics of the email are defined in the feature \text{M0001}.

**Note**

Dynamic actions are only applicable in maintenance operations, not in display functions. Dynamic actions are not executed during batch input processing. Do not set defaults for Q fields of an infotype because the values for these fields are derived from the corresponding P fields.

### 6.2 Organizational Management

Just as described for the PA infotypes, it is also possible to enhance and create new OM infotypes. There are some restrictions with regard to enhancement of OM infotypes (without requiring a modification). Infotypes 1000 (Object) and 1001 (Relationship) cannot be enhanced, since these infotypes are the foundation of the OM module. Typically, enhancements can only be made for OM infotypes that can be maintained directly. Table \text{T777I\_MAINT}\ defines whether maintenance for an OM infotype is allowed.
6.2.1 OM Infotype Enhancement

SAP offers enhancement spot HRBAS00INFTYBL for modifying OM infotypes.

Use Transaction PPCI (see Figure 6.28) for OM infotype development to enhance OM infotype structures or build your own custom OM infotypes. This transaction allows you to maintain and create all the individual components that make up an OM infotype (as always, nnnn is a four-digit unique infotype number) or to extend an existing OM infotype (see Figure 6.29).

These core components make up an OM infotype:

- **Structures:**
  - Pnnnn – Contains the key and header fields for the infotype and the structure serves as the interface between screens and dialog programs using this infotype
  - HRPnnnn – Database table that contains the data fields of the infotype

- **Module pool:** MPnnnn00 with includes MPnnnn20 and MPnnnnBI

- **Screens:**
  - MPnnnn00 1000 (initial) (Note: do not change!)
  - MPnnnn00 2000 (standard)
  - MPnnnn00 3000 (list)
- Graphical user interface: MPnnnn00 with necessary icons and function keys
- Dialog module: RH_INFOTY_nnnn

![Create Infotype](image1)

**Figure 6.28** Create New Infotype via Transaction PPCI

![Enhance Infotype](image2)

**Figure 6.29** Enhance Infotype via Transaction PPCI

Like PA infotype structures, the majority of SAP standard OM infotypes contain an empty include structure CI_Pnnnn contained in database table HRPnnnn that allows adding additional custom fields.

Some infotype structures do not have a CI_Pnnnn structure include, so they can't be enhanced using this approach.

Similar to PA infotype enhancements, the following objects are generated by the system after you have enhanced the infotype structure CI_nnnn:
- New screen ZPnnnn00 0200 with flow logic
- New module pool ZPnnnn00 with the following includes:
  - ZPnnnn10 – Data definitions
  - ZPnnnn20 – Process before output (PBO) routines
  - ZPnnnn30 – Process after input (PAI) routines
  - ZPnnnn40 – Form routines

### 6.2.2 Custom OM Infotypes

If you need to add custom specific fields that cannot be added to an existing OM infotype, you can create a new custom OM infotype via Transaction PPCI. The first step is to create the custom infotype structure HRInnnnn for field infotypes or PTnnnnn for table infotypes. All other objects are automatically created when using Transaction PPCI.

In addition, entries in the main OM infotype configuration tables are made:

- T777D – Infotype Check Table
- T77ID – Enhancements to T777D
- T778T – Infotypes
- T777T – Infotype Texts
- TDCT – Dialog Modules
- T777DITCLASS – Assignment of Check Class

The following tables have to be maintained manually after the OM infotype generation:

- T777I – Infotypes per Object Type
- T777Z – Infotype Time Constraints
- T77CD – Customer-Specific Infotype Settings
- T77NI – Country-Specific Infotypes

Field properties for OM infotypes using the DCIF are defined in table T777IFPROPS and customer changes to the field properties can be made in table T777IFPROPC.
6.2.3 Custom Objects

You can create custom objects in table T778O via IMG path ORGANIZATIONAL MANAGEMENT • BASIC SETTINGS • DATA MODEL ENHANCEMENT • MAINTAIN OBJECT TYPES.

The custom name range for custom object types is 01 to 99. The allowed infotypes per object are defined in table T778T and time constraints are defined in table T778U. Figure 6.30 shows an example of a custom object for defining FUNCTIONAL DEPARTMENT.

![Custom Object 9A–Functional Department](image)

6.2.4 Custom Relationships

Relationships are special subtypes of infotype 1001. Each relationship has two relationship types, indicated by A and B. For each relationship created (A or B), the corresponding inverse relationship is automatically set up. You can create new relationships and their attributes such as time constraints and the object types that are allowed for each relationship and inverse relationship in table T778O via IMG path ORGANIZATIONAL MANAGEMENT • BASIC SETTINGS • DATA MODEL ENHANCEMENT • MAINTAIN OBJECT TYPES.

Figure 6.31 shows the custom relationship Z03 that we have created in order to link our new custom object 9A to an organizational unit (object type = 0), as shown in Figure 6.32, by defining the ALLOWED RELATIONSHIPS.

You can also set up external relationships, which are relationships between external object types (i.e., object types that are not stored in OM).

![Custom Relationship Z03](image)
Setting up evaluation paths can simplify accessing OM data via reports or OM function modules and reduce ABAP code for evaluating complex OM relationships. You can set up custom evaluation paths via IMG path ORGANIZATIONAL MANAGEMENT • BASIC SETTINGS • MAINTAIN EVALUATION PATHS (see Figure 6.33).

The evaluation path can typically be entered as a screen selection criterion in most OM reports. You should consult your OM functional expert about the use of evaluation paths before developing complex ABAP code to select OM relationships.

6.3 Time Management

The PT infotypes in the 2000 to 2999 range cannot be as easily enhanced via Transaction PM01 as the PA infotypes, but in this section we explore options on how to enhance PT infotypes via different techniques and how to make enhancements to CATS.

6.3.1 CATS

Recall that the Cross Application Time Sheet (CATS) is used as a frontend for absence and attendance tracking. This process can have very different flavors and
requirements based on the customer industry and employee types. The standard SAP configuration options for CATS offer a lot of flexibility and allow defining different CATS profile layouts and configuration of different parameters via the IMG. In many cases, the standard CATS configuration options are not sufficient to solve complex business requirements and custom development is required to fill functional gaps. We cover only some samples of the custom development options within CATS in this chapter.

Additional Resources

We recommend *Integrating CATS* by Martin Gillet to learn more information about CATS (SAP PRESS, 2009).

### 6.3.2 Example: Validating CATS Absence/Attendance Entries

Typically, absence/attendance code eligibility is defined via the absence and attendance configuration within the respective IMG section. Only valid absence and attendance codes per configuration can be entered via the CATS. The configuration options are very flexible and in many cases meet the client’s needs. In some cases, though, the business rules can be very complex and the decision criteria and required permissibility rules cannot be reflected by sole use of SAP configuration parameters.

**Requirements**

The following custom requirements for the CATS time entry validation apply to our example:

- Ensure that start and end times are NE ‘0’: If start or end time is EQ ‘0’, create an error message.
- Absences code ‘9999’ is allowed only for employees who are in Employee Group ‘1’ and Employee Subgroup ‘02’: If an employee belongs to another employee group/subgroup combination, create an error message.

**Solution**

To meet the customer requirement, the following solution has been implemented (see Chapter 5, Section 5.1 for more details about customer exits) in Listing 6.2.
1. Enhancement project ZCATS001 has been implemented using SAP enhancement CATS0003.

2. Custom code for custom permissibility checks per customer requirement have been added to include ZXCATU03.

3. Activate ZXCATU03 and function EXIT_SAPLCATS_003 as shown in Figure 6.34.

```
*&------------------------------------------------
*& Include      ZXCATU03
*&------------------------------------------------
*IMPORTING
* VALUE(FIELDS) LIKE CATS_COMM STRUCTURE CATS_COMM
* VALUE(SAP_TCATS) LIKE TCATS STRUCTURE TCATS OPTIONAL
* VALUE(OLD_DATA) TYPE BOOLEAN OPTIONAL
* TABLES
* I_MESSAGES STRUCTURE CATS_MESG
*&------------------------------------------------
DATA: fdate(10).
DATA:
i0001 TYPE TABLE OF p0001 INITIAL SIZE 0
     WITH HEADER LINE.
* check for missing Start time or End time
IF fields-beguz NE space AND
fields-enduz EQ space.
```

Figure 6.34  CATS Customer Exit EXIT_SAPLCATS_003
i_messages-msgid = 'Z_HR'.
i_messages-msgty = 'E'.
i_messages-msgno = '908'.
i_messages-pernr = fields-pernr.
i_messages-catsdate = fields-workdate.
WRITE fields-workdate TO fdate MM/DD/YYYY.
i_messages-msgv1+26 = fdate. CLEAR fdate.
CONDENSE i_messages-msgv1.
APPEND i_messages. CLEAR i_messages.
ELSEIF fields-beguz EQ space AND
    fields-enduz NE space.
i_messages-msgid = 'Z_HR'.
i_messages-msgty = 'E'.
i_messages-msgno = '907'.
i_messages-pernr = fields-pernr.
i_messages-catsdate = fields-workdate.
WRITE fields-workdate TO fdate MM/DD/YYYY.
i_messages-msgv1+26 = fdate. CLEAR fdate.
CONDENSE i_messages-msgv1.
APPEND i_messages. CLEAR i_messages.
ENDIF.

* Get infotype 0001 information
CLEAR: i0001, fdate.
CALL FUNCTION 'HR_READ_INFOTYPE'
  EXPORTING
    pernr      = fields-pernr
    infty      = '0001'
  TABLES
    infty_tab    = i0001
EXCEPTIONS
    infty_not_found = 1
    OTHERS     = 2.

* Absence Eligibility Checks
IF sy-subrc = 0.
  LOOP AT i0001 WHERE pernr EQ fields-pernr AND
    begda LE fields-workdate AND
    endda GE fields-workdate.
      EXIT.
  ENDLOOP.
ENDIF.
WRITE fields-workdate TO fdate MM/DD/YYYY.
CASE fields-awart.
  WHEN '9000'.
IF NOT ( i0001-persg EQ '1' AND i0001-persk EQ '02' ).
ELSE.
   i_messages-msgid = 'Z_HR'.
   i_messages-msgty = 'E'.
   i_messages-msgno = '904'.
   i_messages-pernr = fields-pernr.
   i_messages-catsdate = fields-workdate.
   i_messages-msgv1 = fields-awart.
   i_messages-msgv2 = fdate.
   i_messages-msgv3 = i0001-persg.
   i_messages-msgv4 = i0001-persk.
   APPEND i_messages. CLEAR i_messages.
ENDIF.
WHEN Others.
ENDCASE.

Listing 6.2 Code Sample for INCLUDE ZXCATU003

6.3.3 Enhancing Time Infotypes

Time infotypes cannot be enhanced via the standard SAP infotype enhancement concept using PM01. Any type of custom change has to be applied using one of the following three techniques:

- Implement one of the methods in BAdI HRPA00INFTY.
  - BEFORE_OUTPUT
  - AFTER_INPUT
  - IN_UPDATE
- Use customer exit PBAS0001.
  - EXIT_SAPFP50M_001
  - EXIT_SAPFP50M_002
- Use an existing enhancement spot in one of the includes of module pool MP200000 close to the area that you want to enhance.

Requirements

The customer requirement is to check the permissibility of a specific absence code 9000 by employee group subgroup combination. SAP configuration does not allow specifying this restriction.
Solution 01: Use BAdI HRPAD00INFTY

Implement the **AFTER_INPUT** method (see Figure 6.37) in BAdI HRPAD00INFTY (as shown in Figure 6.35 and Figure 6.36) to check for the employee group/subgroup combination when infotype 2001 is entered; create an error message if absence code ‘9000’ is used for employees that are not assigned to specified Employee Group ‘1’ and Employee Subgroup ‘02’. See Listing 6.3.

![Figure 6.35 BAdI HRPAD00INFTY](image)

![Business Add-Ins: Display Definition HRPAD00INFTY](image)

![Business Add-In Builder: Change Implementation Z_2001_CHECK](image)

Figure 6.35 BAdI HRPAD00INFTY

Figure 6.36 Implement BAdI via Transaction SE19
Figure 6.37  Custom Code for AFTER_INPUT Method

method IF_EX_HRPAD00INFTY-AFTER_INPUT.
DATA: i0001 TYPE TABLE OF p0001,
w0001 TYPE p0001.
CASE new_innnn-infty.
  WHEN '2001'.
    ASSIGN new_innn TO <p2001> CASTING.
    IF <p2001>-subty = '9000'.
      CALL FUNCTION 'HR_READ_INFOTYPE'
      EXPORTING
        pernr = <p2001>-pernr
        infty = '0001'
      TABLES
        infty_tab = i0001[].
      EXCEPTIONS
        infty_not_found = 1
        OTHERS = 2.
    ELSE.
      MESSAGE e999 [ychr].
      ENDIF.
    ENDIF.
  ELSE.
    ENDCASE.
ENDMETHOD.
EXCEPTIONS
   infy_not_found = 1
   OTHERS      = 2.
IF sy-subrc = 0.
  LOOP AT i0001 INTO w0001
    WHERE pernr = <P2001>-pernr
    AND begda <= <P2001>-begda
    AND endda >= <P2001>-enda.
    EXIT.
  ENDLOOP.
ENDIF.
IF sy-subrc = 0.
  IF w0001-persG EQ '1' and
     w0001-persK EQ '02'.
    ELSE.
      MESSAGE e999(ythr).
    ENDIF.
  endif.
ENDIF.
ENDCASE.
endmethod.

Listing 6.3 Code Sample for AFTER_INPUT Method

Solution 02: Use Customer Exit PBAS0001

Implement customer exit PBAS0001 and create ABAP code in include ZXPADU02, which is used in function module EXIT_SAPFP50M_002 (as shown in Figure 6.38) for after input processing to read infotype 0001 in order to check the employee group/subgroup when an infotype 2001 with subtype = ‘9000’ is maintained, similar to the code in the BAdI class.

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Be sure to always check for the specific infotype via a CASE statement when using generic BAdIs, methods, or customer exits that are executed for every infotype. See SAP Note 897623 (User Exits and BAdIs for Time Management) for more information.</td>
</tr>
</tbody>
</table>
The core “engine” for payroll calculation is \texttt{RPCALCn0}, where \( n \) stands for the specific one-byte country indicator defined in table \texttt{T500L}. The US payroll driver program is \texttt{RPCALCU0}. For time evaluation, the country-independent driver program is \texttt{RPTIME00}. The driver programs do not contain any hardcoded business rules; they access the business logic via so-called schemas and personnel calculation rules (PCRs, and formerly also called cycles). The business rules are defined within a schema in form of functions and PCRs. The personnel calculation rules are further defined by operations. A schema is processed sequentially by the driver program, and the logic of functions and operations can be further influenced by SAP time or payroll configuration, as illustrated in Figure 6.39. The overall time evaluation logic is similar to the payroll calculation logic shown in Figure 6.39, the only difference being that the time evaluation calculation is started using \texttt{RPTIME00} instead of \texttt{RPCALCn0} and that the time evaluation driver is accessing different configuration tables and employee data.
Functions and operations are represented by dedicated ABAP form routines that define the logic to be executed when a function or operation is called with certain defined parameters from the driver program.

![Figure 6.39 Payroll Calculation Logic](image)

The payroll driver reads specific master data and payroll infotypes and processes them according to the payroll configuration and business logic defined in the payroll schemas and payroll personnel calculations rules. Payroll processes wage types and amounts in the country-specific currency. The output of the payroll driver is stored in cluster tables within PCL2, and the cluster ID Rn is dependent on the country version where n stand for the country indicator. The cluster ID for the US is therefore RU.
The time evaluation driver processes time types, but does not process any monetary amounts—just hours based on time entries and employee work schedule information. The output of the time evaluation processing is stored in cluster tables, where PLC2 cluster B2 contains the most important output values in form of wage types, time types and hours per employee, and date. The output of time evaluation can be directly processed via SAP payroll RPCALCn0 or the data can be extracted from PCL2 to be interfaced to third-party payroll solutions.

SAP delivers a set of sample time and payroll schemas and PCRs, but typically client-specific schemas and PCRs have to be built to reflect the custom business requirements for time and payroll calculations. Schemas are maintained via Transaction PE01 (Schema Editor, as shown in Figure 6.40) and PCRs via Transaction PE02 (Rule Editor, as shown in Figure 6.41).

---

**Display Schema : TM00**

<table>
<thead>
<tr>
<th>Line</th>
<th>Func.</th>
<th>Par1</th>
<th>Par2</th>
<th>Par3</th>
<th>Par4</th>
<th>O Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>01010</td>
<td>CMN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Time Evaluation With Clock Times</td>
</tr>
<tr>
<td>01320</td>
<td>CMN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Evaluation of time data specifying</td>
</tr>
<tr>
<td>01380</td>
<td>**</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>clock times (including personnel</td>
</tr>
<tr>
<td>01390</td>
<td>**</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>time events or FBC times)</td>
</tr>
<tr>
<td>01690</td>
<td>**</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>initialization</td>
</tr>
<tr>
<td>01680</td>
<td>MDD</td>
<td>MOST GEN</td>
<td></td>
<td></td>
<td></td>
<td>Define groupings</td>
</tr>
<tr>
<td>01690</td>
<td>EINI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>End of processing block</td>
</tr>
<tr>
<td>01700</td>
<td>**</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Day processing</td>
</tr>
<tr>
<td>01700</td>
<td>**</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Options:</td>
</tr>
<tr>
<td>01710</td>
<td>CHECK</td>
<td>RHR</td>
<td></td>
<td></td>
<td></td>
<td>Set regular acc for payroll, if required</td>
</tr>
<tr>
<td>01710</td>
<td>CHECK</td>
<td>FUT</td>
<td></td>
<td></td>
<td></td>
<td>Allow evaluations for future periods</td>
</tr>
<tr>
<td>01720</td>
<td>DGR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Process reduced hours</td>
</tr>
<tr>
<td>01750</td>
<td>OPTT</td>
<td>MAIL 1</td>
<td></td>
<td></td>
<td></td>
<td>* Mail to administrator if errors occur</td>
</tr>
<tr>
<td>01750</td>
<td>**</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01760</td>
<td>BLOCK</td>
<td>BES</td>
<td></td>
<td></td>
<td></td>
<td>Provide time data</td>
</tr>
<tr>
<td>01770</td>
<td>IF</td>
<td>NOT SCM</td>
<td></td>
<td></td>
<td></td>
<td>No simulation for future periods</td>
</tr>
<tr>
<td>01780</td>
<td>FERT</td>
<td>1020</td>
<td></td>
<td></td>
<td></td>
<td>Evaluate errors from pair formation</td>
</tr>
<tr>
<td>01790</td>
<td>P011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide time pairs and daily WGS</td>
</tr>
<tr>
<td>01800</td>
<td>ACTION</td>
<td>1010</td>
<td></td>
<td></td>
<td></td>
<td>End if daily work schedule still active</td>
</tr>
<tr>
<td>01810</td>
<td>A003</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Process work center substitution</td>
</tr>
<tr>
<td>01840</td>
<td>ACTION</td>
<td>1009 AB</td>
<td></td>
<td></td>
<td></td>
<td>Convert daily WGS if RW=1 and leave</td>
</tr>
<tr>
<td>01850</td>
<td>P011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide absence data of the day</td>
</tr>
<tr>
<td>01860</td>
<td>P002</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide attendance data of the day</td>
</tr>
<tr>
<td>01870</td>
<td>PITF</td>
<td>1009 GEN</td>
<td></td>
<td></td>
<td></td>
<td>Process absence/attendance reasons</td>
</tr>
<tr>
<td>01880</td>
<td>ACTION</td>
<td>1009</td>
<td></td>
<td></td>
<td></td>
<td>Process generated locked records</td>
</tr>
<tr>
<td>01890</td>
<td>P005</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>* Provide overtime data of the day</td>
</tr>
<tr>
<td>01900</td>
<td>PITF</td>
<td>1040 GEN</td>
<td></td>
<td></td>
<td></td>
<td>Set FTPTYPE for overtime</td>
</tr>
</tbody>
</table>

*Figure 6.40 Schema TM00–Time Evaluation with Personnel Time Events*
Enhancing Applications Using SAP Tools

PCRs can be edited in table or structural graphics format (as shown in Figure 6.42). This book does not cover the details of the configuration for payroll or time evaluation. Rather, we provide insight into how SAP standard payroll or time evaluation tools can be enhanced if complex custom requirements cannot be implemented using SAP standard operations and functions.

SAP functions and operations are maintained via Transaction PE04 (see Figure 6.43). New functions or operations can be created or existing ones can be enhanced if
required. Functions and operations are nothing more than ABAP form routines that are defined within dedicated includes of the respective driver program.

![Maintain Functions and Operations](image)

**Figure 6.43** Transaction PE04 Selection Screen

Each function is defined by a specific form routine with the naming convention FUnnnnn, where nnnnn stands for the five-byte function name that can be entered in the schema editor. Figure 6.44 shows an example of function P2006. Each operation is defined by a specific form routine with the naming convention OPnnnnnn, where nnnnnn stands for the five-byte operation name that can be entered in the Rule Editor.

The ABAP code for payroll operations and functions can be found in the following includes:

- **PCFDCZnn0** – Include for custom data, infotype and table definitions
- **PCBURZnn0** – Include for custom operations and functions, where nn stands for the two-byte country-specific ISO code that is defined in table T500L; for the US payroll driver RPCALCU0, the includes are RCFDCZUS0 and PCBURZUS0

If Concurrent Employment (CE) is used, the includes for RPCALCnn0_CE are PCFDCZnnn0_CE and PCBURZnnn0_CE.

The ABAP code for time management functions can be found in report SAPFP51T. Customer-specific time management functions and operations are stored in the include RPTMOZ00.
The following steps have to be followed to create a new function via Transaction PE04:

1. Create new operation/function.
2. Define country assignment (usually time functions are international).
3. Define parameters.
4. Define parameter values.
5. Define infotypes to be used.

**Note**

You should use only digits, letters A-Z, and the following special characters for custom functions and operations: & ( ) - ; < >

This only applies for new objects to be created; payroll schemas and rules that already exist can be processed further. For more information, see SAP Note 203490.
In older SAP versions, you needed a developer key to create the custom code for new operations or functions in the time or payroll includes. With the new enhancement spot concept, you can now implement the new code as an enhancement within these includes without requiring an access code or creating a repair.

We now illustrate the steps required to create a new function using a new time evaluation function as an example. The steps to create payroll and time operations and functions are very similar and the following steps can be applied.

**Example: Reading Infotype 0208 (Work Tax Area) in Time Evaluation**

Our example illustrates a real-life example and, after providing the high level customer requirements, we walk you through the steps of our enhancement solution.

**Requirement**

In our sample company, we have a requirement to make certain business rules dependent on the infotype 0208 (Work Tax Area). For example, in California (which is defined as work tax area 1000), there is a different overtime rule that should be applied in time evaluation. It is only for employees who work in California and who are assigned to work tax area 1000 in infotype 0208.

**Solution**

In standard SAP, you cannot access infotype 0208 from within a time evaluation schema since there is no standard function or operation available to read it. We will develop a custom time operation \_WTAR to allow us to read this infotype within a time evaluation PCR in order to make further decisions on the work location of the employee based on infotype 0208. Follow along in Figure 6.45 through Figure 6.50.

With parameter setting \_WTARxxxx the new custom operation will query whether work area xxxx exists in infotype 0208 on the day currently processed in time evaluation and provide a return value of either ‘Y’ or ‘N’ in the variable argument.

Once we have developed this new operation, we will use it later on in our custom personnel calculation rule -O12 (see Figure 6.50); we’ll create -O12 to evaluate the return value from the variable argument in our custom PCR to make decisions based on this return value for the custom overtime calculation rules.

With parameter setting \_WTAR we can check whether an infotype record 0208 exists at all.
We should also create a custom error message in table T55SE via Transaction SM31 view V_T55SE for time evaluation MODBE to be able to react to missing infotype 0208 authorization and write out an error message.

**Figure 6.45** PE04 Create Custom Operation _WTAR

**Figure 6.46** Assign Countries and Define Parameters
After displaying the include RPTMOZ00 via Transaction SE38, you can show all enhancement spots via the Enhance button (Shift) + (F4). Select an enhancement spot and go to Edit • Enhancement Options • Create Implementation.

After creating the enhancement implementation in RPTMOZ00, we add the code in Listing 6.4 between the Enhancement and EndEnhancement events (Figure 6.49).
form op_wtar.
  data: subrc like sy-subrc.
  statics p0208 type p0208 occurs 2
     with header line.
  loop at p0208 where pernr = pernr-pernr.
  exit.
  endloop.
  if sy-subrc ne 0.
    clear: p0208[], p0208.
    call function 'HR_READ_INFOTYPE'
       exporting
          pernr = pernr-pernr
          infty = '0208'
          begda = '18000101'
          endda = '99991231'
       importing
          subrc = subrc
          tables
            infty_tab = p0208
       exceptions
          infty_not_found = 1
          others = 2.
* handling missing IT0208 authorization
  if ( subrc = 4 ) or ( subrc = 12 ).
    perform col_error4 using acdate 'ZW'       space.
    check error_status ne '2'.
  endif.
  endif.
  case op10+5(4).
    when space.
      loop at p0208 where begda le acdate and endda ge acdate.
      exit.
    endloop.
    if sy-subrc eq 0.
      vargt = 'Y'.
    else.
      vargt = 'N'.
    endif.
  when others.
* does IT0208 exist in general?
  loop at p0208 where begda le acdate and endda ge acdate
  and wtart eq op10+5(4).
Listing 6.4 Code Sample for Custom Operation _WATR

```abap
exit.
endloop.
if sy-subrc eq 0.
  vargt = 'Y'.
else.
  vargt = 'N'.
endif.
endcase.
perform fillvargt.
endform.
```

After creating the enhancement code for the new operation, spell-check the code and activate both the code and your enhancement. Now you should be ready to use your new custom operation within custom time evaluation rules to make decisions based on the employee tax area infotype 0208, as shown in Figure 6.50.
In the example shown in Figure 6.50, we edit custom rule -O12 via the rule editor transaction, Transaction PE02. Our new custom operation _WTAR can now be used. We add it in our custom rule -O12 to check whether the work tax area is 1000 for the currently processed employee. If the return parameter in the variable argument is EQ "Y", personnel calculation rule -O99 is executed for further processing and calculating the CA-specific overtime. The example illustrates how you can solve a custom requirement by creating a custom time management operation.

We realize that there are other options to determine the employee state for overtime calculation—for example, by checking the infotype 0001 personnel area or personnel subarea field via standard operation OUTWP using parameter PLANT or PLTSC—but in this specific client scenario, it was required to check the work tax area.

The technical concept and approach for developing custom functions and operations for payroll and time management is the same; the example gives you a guideline for how to approach a similar development request in both functional areas.

If you need to debug the code of an existing or new function or operation, you can use one of the following methods:

- Set a breakpoint in the schema via function BREAK.
- Set a breakpoint in the PCR via operation BREAK using user parameter AB4.
- Set a breakpoint in the form routine for a specific function or operation.
- Enter /h into the command box before starting RPCALCn0 or RPTIME00 and then go to Set Breakpoints • Breakpoint At • Subroutine and enter the program name (RPCALCn0 or RPTIME00) and the subroutine name (FUnnnnn for a function or OPnnpnnnn for an operation).
You will be able to look at all the temporary tables, data, and intermediate results during run time for `RPTIME00` or `RPCALCNX`. Just make sure to switch the time or payroll log off and use only a small population (ideally one employee) and a small timeframe for your debugging troubleshooting scenario in order to pinpoint the exact section in the code you are troubleshooting.

Note

Please also see the “HR Experts” whitepaper *When and How to Develop Custom Functions and Operations* by Steve Bogner.

### 6.5 Summary

The SAP enhancement technology and tools that come with a standard-delivered SAP system—especially the SAP ERP HCM-specific tools—are almost limitless and should allow you to meet even the most complex custom requirements. The techniques described in this chapter for infotype enhancements, features, dynamic actions, and development of custom operations and functions will enable you to help design and implement good solutions for filling even the most complex functional HCM gaps using SAP standard tools.

As we have shown through the examples, there can be several different technical solutions for resolving one specific customer requirement. You should always explore the option of a low-impact solution—ideally one via standard configuration—and always consider long-term implications and impacts on other processes, programs, and future changes. Try to design your solution to be as flexible as possible and ideally drive decisions based on tables or features so that you do not have to “hard code” plausibility checks and can adjust to business process changes rapidly—ideally, without the need for ABAP development.

Having explored enhancement and development techniques for enhancing the existing HCM application in this chapter, we now turn our attention to SAP ERP HCM-specific reporting tools.
Index

A

ABAP custom development, 660
ABAP event
   END-OF-SELECTION, 411
   GET GROUP, 156, 344
   GET OBJEC, 160
   GET PAYROLL, 154, 356, 358, 379, 408
   GET PERAS, 156, 157, 344, 348, 356
   GET PERNR, 155, 156, 169, 344, 348, 352, 356, 385, 408
   GET PERSON, 156, 344, 348
   INITIALIZATION, 369, 589
   START-OF-SELECTION, 352, 360, 590
ABAP List Viewer (ALV), 343, 388, 404, 666, 671
   column header, 670
   default functions, 669
   DISPLAY, 668, 669
   display variant, 405
   FACTORY, 666
   field definitions, 389
   function modules, 403
   grid, 389
   list, 412
   list layout, 412
   list output, 413
   output list, 388
   output pattern, 669
   sorting, 670
   table layout, 404
   totaling, 670
ABAP-OO class, 243
ABAP statement
   AUTHORITY-CHECK, 598, 599
   CALL FUNCTION, 662
   CALL METHOD, 208
   CASE, 161, 228
   DEFINE, 172
   DO VARYING, 352, 353
   ENHANCEMENT, 214, 273
   GET, 147, 153
   ABAP statement (Cont.)
   IF, 228
   INFOTYPE, 157, 385
   INFOTYPES, 153, 157, 385, 162, 344, 361, 590
   LOOP, 153, 379, 409
   MESSAGE, 174
   METHOD, 206
   NODES, 344, 357
   PARAMETERS, 162, 405
   PROVIDE, 153, 160, 348, 361, 362
   PUT, 147
   SELECT, 161, 182, 345, 590
   SELECTION-OPTIONS, 163
   TABLES, 153, 159, 358, 367, 368
   TYPE, 353
   TYPE-POOLS, 389
   UPDATE, 163
   WAIT, 664
   WRITE, 388
Absence Quota, 217, 387
Absence Quota Compensation, 217
Access authorization, 237
Adding customer-specific fields, 687
Add-on, 609
Ad Hoc Query, 304
Administrative fields, 82
Administrative Services, 611, 612
Administrator for HR Master Data, 569
Administrator for Time Recording, 569
Administrator group, 569
Adobe, 607
Adobe Flash Island, 640
ALE, 499, 501, 504, 514, 631
   application layer, 499
   BAPI usage, 503, 504
   communication layer, 499
   distribution layer, 499, 502
   distribution model, 502, 503, 504
   generate interface, 502
   logical system, 503
   synchronous, 499
Allowed relationships, 256
API, 696
Applicant data, 571
Applicant group, 572
Applicant range, 572
Application, 63
APPLICATION AREA, 196
Application-to-Application Navigation, 632
Appraisals, 66
Approval, 598
Area Identifier for Cluster ID, 572
Assignment of check class, 255
asXML, 511, 512
Asymmetrical verification, 577
  Asymmetrical double verification, 572
Attendance quota, 387
Audit trail information, 364
Authorization, 579, 587, 589, 590
Authorization check, 565, 566, 570, 571, 572,
  574, 579, 583, 584, 588, 589, 591, 598,
  600, 602
Authorization concept, 559
Authorization enhancement, 590
Authorization field, 559, 560, 561, 568, 594
Authorization for batch input sessions, 597
Authorization level, 560, 568, 570, 571, 572,
  576
Authorization main switch, 568, 569, 571,
  586, 596, 598
Authorization object, 559, 560, 564, 565,
  568, 569, 570, 571, 572, 585, 593, 594
  P_ABAP, 573, 574, 575
  P_APL, 571
  P_DEL_PERNR, 695
  PLOG, 576
  P_NNNNN, 570, 572, 587
  P_NNNNCON, 585, 588
  P_ORGIN, 560, 567, 571, 572, 574, 575,
    576, 585, 586, 587, 598
  P_ORGINC, 568, 585, 586, 588
  P_ORGXX, 569, 572, 576, 585, 587
  P_ORGXXCON, 569, 585, 588
  P_PCLX, 572, 573
  P_PERNR, 571, 572, 587, 598
  P_TCODE, 567
  S_BDC_MONI, 597
  Authorization object (Cont.)
    S_SPO_ACT, 597
    S_TCODE, 567
Authorization profile, 560, 564, 568, 569,
  579, 581, 582, 588
Authorization role
  SAP_EMPLOYEE_ESS_WDA_1, 621
  SAP_EMPLOYEE_ESS_WDA_2, 621
  SAP_MANAGER_MSS_NWBC_2, 628
  SAP_MANAGER_MSS_SR_NWBC_2, 631
Authorization switch, 587, 589, 590

B
Background Job Creation, 597
BAdI enhancement
  HR_COUNTRY_FILTER, 624
Bank transfer, 377
BAPI Explorer, 475
legend, 476
Batch data communication (BDC), 523, 524
  background execution, 531, 532
  create session, 656
  foreground execution, 531
  parallel processing, 531
  recording, 528
  server destination, 531
  session, 523, 524, 525, 527, 530, 532,
    533, 553, 595
  session processing, 529
Batch input, 516, 517, 521
Batch input session, 523
Batch Input: Session Overview, 529, 532, 553
  analysis, 530
  process, 531
Batch-input authorizations, 597
BDCDATA, 519, 521, 522, 523, 524, 526
Blueprint phase, 193
Breakpoint, 164
Budgeting, 67
Business Add-In (BAdI), 164, 204, 209, 230,
  590, 591, 592, 594, 679, 683
  HRBAS00_GET_PROFILE, 593
  HRBAS00_STRUAUTH, 592
  HRESS_MENU, 624
  HRHAP00_FOLLOW_UP, 72
Business Add-In (BAdI) (Cont.)
HRPAD00AUTH_CHECK, 591, 594
HRPAD00AUTH_TIME, 593
HRPAD00CHECK_AUTH, 593
HRPAD00_ENTRY_LEAVE, 177
HRPAD00INFTY, 222, 243, 261
HRPAD00INFTYDB, 243
HRPAD00INFTYPE, 230
HRPAYXX_DELETE_PERNR, 695
HRSFI_BEMPL_DATA_REPLICATION, 684, 685, 687, 688
HRSFI_B_ENHANCE_EMPL_DATA_REP, 688
HRSFI_B_FILE_NAME, 687
HRSFI_B_LEADING_CONTRACT, 686
HRSFI_COMP_ALT_SALARY_ANNPCT, 693
HRSFI_COMP_ALT_SALARY_ANNUAL, 693
HRSFI_COMP_STD_SALARY_FIELDS, 692
HRXSS_SERVICE_FILTER, 624
XSS_DATAORIGIN, 631
Business Application Interface (BAPI), 164, 183, 474, 475, 476, 478, 480, 482, 503, 506, 510
execution, 479
BAPI_EMPLOCOMM_CREATE, 180, 183
HR_EMPLOYEE_DEQUEUE, 178
HR_EMPLOYEE_ENQUEUE, 178
remote execution, 479
testing, 476, 477, 478
Business Document Server, 440
Business function, 469, 472, 475, 514, 688
access layer, 470
ADDREMPUS, 473
BUS1065, 471, 472, 473
display, 473
EmployeePrivateAdrUS, 475
EMPLOYEEET, 473
EMPLOYEEUS, 473
Business object (Cont.)
capsulation, 470
inheritance, 473
instance-dependent, 476, 478
instance-independent, 476, 478
instantiated, 476, 479
integrity layer, 470
interface layer, 470, 474
kernel, 470
methods, 474
type, 471
Business Object Repository Browser, 471, 475, 476, 632
legend, 473
methods, 474
Business service pages (BSP), 62

C

CALL FUNCTION
performing on end of task, 663
starting new task, 662, 663
CALL TRANSACTION, 521, 523, 528
CTU_PARAMS, 522
messages, 523
mode, 522
synchronous, 522
update, 522
Candidate, 63
Candidate Attribute infotypes, 64
CD manager, 370
Central person, 348
Change documents, 185
Change locked data records, 576
Check class, 240, 243
Check Class for PA infotypes, 96
Check indicator, 566
CHIP, 617
Classes, 164, 389
CL_EXM_IM_HRBAS00_STRUAUTH, 593
CL_HRBAS_INTY_CONTAINER, 118
CL_HRPAD00AUTH_CHECK_STD, 592
CL_HRPA_INFOTYPE_0002, 98
CL_HRPA_INFOTYPE_nnnn, 95, 100, 243
CL_HRPA_TEXT_CLUSTER, 185
ZCL_HRPA_INFOTYPE_9nnn, 100
Index

Classic output, 367
Cloud integration, 674
Cloud services, 697
Cloud solutions, 61
Cloud system, 675
Cluster 99, 356
Cluster definition, 380
Cluster directory, 137, 370, 373, 377
Cluster ID, 123, 369, 370
  B1, 124, 387
  B2, 370, 387, 425
  CD, 370
  G1, 124
  L1, 124
  LA, 572
  LB, 572
  PC, 124
  PCL1, 124
  PS, 370
  PT, 370
  RD, 356
  RU, 356, 371, 377, 379
  RX, 370, 383
  TC, 124
  TX, 124, 572
  TY, 572
  UN, 370
  ZI, 124
  ZL, 371
Cluster key, 372
Cluster table, 119, 137, 184
  PCL1, 123, 124, 142, 185, 237, 387, 572
  PCL2, 59, 123, 137, 138, 142, 198, 266,
    356, 369, 371, 387
  PCL3, 142, 144
  PCL4, 142, 144, 185, 339, 340, 364, 365,
    368, 371, 387
  PCL5, 142, 144
Compensation data integration, 674, 677, 689
Compensation planning, 67
Compensation review, 67
Compensation scenario, 690
Composite role, 564
Concurrent Employment (CE), 147, 156, 344,
  623, 685, 686
Configuration tables, 78

Context-based authorization, 584, 586
  check, 569, 570, 568
  objects, 584, 585, 588
Contraction, 361, 362
Control indicator, 374
Controller, 95
Conversion
  catalog legacy data, 546
  cutover plan, 556
  data cleanup, 546
  external number range, 544, 552
  maintain number range, 544
  payroll balances, 554
  personnel number range, 546
  preparation, 544
  pre-processor, 549, 552
  testing, 555
Conversion class, 97
Corbu, 616
Core night work, 388
Cost center, 101
Country groupings, 357, 561
Country-specific infotypes, 255
Country versions, 370
Cross-Application Timesheet, 47, 129, 257
  CATS Classic, 49
  CATS profile authorization groups, 598
  CATS profiles, 129
  CATS Regular, 49
CSV, 678, 682
Cumulated time balances, 388
Cumulated variable, 388
Current results, 374
Custom infotype, 233
  OM infotype, 255
Custom objects, 256
Custom operation, 275
Custom payroll country versions, 383
Custom relationship, 256
Customer exit, 194, 232
  HRRPAI01, 177
  PBAS0001, 264
Customer-specific authorization check, 571,
  591
Customer-specific authorization object, 594,
  595
Customer-specific implementation, 684
Index

D

Daily Time balances, 388
Data Browser, 280, 281, 282, 283, 284, 285
Data conversion
  preparation, 546
Data integration, 679
Data migration, 515, 533
  preparation, 544
Data selection period, 153, 345, 359
Data status, 367
Date and time calculations, 356
Debugging operations, 187
Decision operations, 187
Decoupled Infotype Framework (DCIF), 78, 94, 242
Degree of simplification, 573, 575
Delimited, 82
Dell Boomi, 696
Depth, 582
Determining the leading assignment of a central person, 686
Dialog module, 92, 223, 238, 254, 517, 518
  RH_INFOTYPE_1005, 117
  RP_9002, 239
Digital Personnel File, 607
Direct deposit, 377
Direct input, 515, 516, 549
Display authorization data, 601
Display depth, 367
Display Payroll Results, 141
Distributed processing, 653, 671
DOM, 510
Domain, 224
Donor country solution, 383
Double verification principle, 559, 576
Dynamic action, 245, 248
Dynamic selection, 348

E

ECC 4.6, 389
Eclipse tools, 695
EHP 4, 63, 73
EHP 5, 63, 72
Email address, 687
Employee, 580
Employee Central, 673, 679
Employee data, 677
Employee group, 38, 560, 568, 685
Employee group/subgroup, 579
Employee Interaction Center, 606, 607, 611
Employee master data, 677
Employee organizational assignment, 568
Employee Self-Service (ESS), 65, 584, 598, 601, 605, 606, 615, 619, 645
  authorizations, 621
  BAdI enhancement, 622
  business functions, 619
  business package, 621
  customizations, 623
  data tracking, 625, 626
  Homepage Framework, 618, 620
  launchpad, 621
  on behalf of employees, 627
  service settings, 626
  services menu, 624
  unavailability of services, 624
  Web Dynpro for ABAP, 618, 621, 623, 624, 626
Employee subgroup, 38, 197, 201, 560, 568, 685
Employment percentage, 693
Enhancement, 198
  CATS0003, 259
  PBAS0001, 227
  framework, 612
  implementation, 273
  option, 222, 684
Enhancement package
  activation, 612
  business function, 193, 609, 610, 611, 612
  business function set, 610
  switch, 610
  technical usage, 610
Enhancement points, 195, 212
Enhancement project, 198
ENHANCEMENT-SECTION, 213
Enhancement spot
  HRBAS00INFTYBL, 253
  HRPAD00INFTYDB, 243
  HRPAD00INFTYUI, 243
Enterprise business functions, 612
Enterprise Compensation Management (ECM), 67, 612, 673
Enterprise extensions, 612
Enterprise Learning, 72
    infotypes, 74
Enterprise Services Repository (ESR), 688
Enterprise structure, 38
E-Recruiting, 62
    objects, 64
Error operations, 187
Evaluation path, 257, 367, 368, 580, 581, 582, 624
Event, 344, 356
Event handler, 389
External infotypes, 114
External object types, 582
Extraction of employee data for SuccessFactors, 679

Fit-gap analysis, 193
Form Builder, 417, 466
    activate form, 453, 455
    attributes, 438
    folder node, 444
    Form Painter, 437
    generated function module, 439
    global data, 439
    graphic, 442
    graphic import, 443
    interface, 439
    language setting, 438
    layout, 436
    line types, 452
    logo, 440, 442, 443
    loop node, 445, 447
    main window, 449, 451
    next page, 441
    output options, 438
    page, 440
    print program, 453
    print program execution, 454
    print program test, 453, 455
    program lines node, 446, 447
    properties, 440
    routines, 440
    secondary window, 443, 449
    Smart Style, 439
    table line types, 451
    table node, 450, 451
    table node calculations, 452
    table node cell, 453
    table node row, 453
    table node width, 452
    text node, 446, 453
    window, 440
For-period, 372
Full localization, 383
Function, 59, 265
    BREAK, 276
    EXPRT, 332
    P0167, 59
    P2006, 269
    PIT, 60
    QUOTA, 198
    TRANS, 555

F

Fast Entry for Master Data, 655
    create from proposal, 658, 658
    locked records, 657
    menu, 656
    operation indicator, 658
    personnel entry, 657
Feature, 176, 186, 244, 561
    ADDR, 458
    BIMAP, 597
    ENTRY, 176
    OCHRF, 461
    OTSEL, 495
    OTVSN, 495
    PBCHO, 460
    TMSTA, 186
    VDSK1, 561
Field catalog, 388
Field symbol, 665
File name, 687
FILE NAME for FILE, 687
File path, 685, 687
File transfer, 679, 682
Filling of ALV table, 389
Functional gap, 193

Function exit
- EXIT_SAPF50M_001, 227
- EXIT_SAPF50M_002, 228
- EXIT_SAPFP50M_002, 264
- EXIT_SAPLCATS_003, 259

Function groups, 174

Function module, 164, 173, 184, 352, 389, 582, 583
- AUTHORITY_CHECK, 599, 600
- BAPI_EMPLOYEE_DEQUEUE, 171
- BAPI_EMPLOYEE_ENQUEUE, 171
- BAPI_EMPLOYEET_DEQUEUE, 479
- BAPI_EMPLOYEET_ENQUEUE, 479
- BAPI_TMEQUOTA_GETDETAILEDLIST, 387
- BDC_CLOSE_GROUP, 525
- BDC_INSERT, 524
- BDC_OPEN_GROUP, 524
- CD_EVALUATION_PERIODS, 374
- CD_GET_INFO, 376
- CD_HIGHEST_PAYDT, 376
- CD_PAYROLL_UNTIL, 376
- CD_READ_LAST, 376, 377
- CD_READ_PREVIOUS, 375
- CD_RETCALC_PERIOD, 376
- CU_READ_RGDIR, 373, 377
- CU_READ_RGDIR_NEW, 380
- HR_ALV_LIST_DISPLAY, 413
- HR_CHECK_AUTHORITY_INFTY, 590
- HR_COMPUTE_YEARS_BETWEEN_DATES, 356
- HR_ENTRY_DATE, 175
- HR_FEATURE_BACKFIELD, 190, 245
- HR_FEATURE_BACKTABLE, 190
- HR_FORMS_TIM_GET_B2_RESULTS, 387
- HR_GET_TIME_BETWEEN_DATES, 356
- HR_INFOTYPE_LOG_GET_DETAIL, 185, 364
- HR_INFOTYPE_LOG_GET_LIST, 185, 364
- HR_INFOTYPE_OPERATION, 178
- HR_READ_INFOTYP, 590
- HR_READ_INFOTYPES, 661
- HR_READ_TIMEDATA_PSP, 211, 385, 386
- HR_TIME_RESULTS_GET, 387
- HRWPC_GET_EMPLOYEE_NAME, 351

Function module (Cont.)
- PYXX_GET_EVALUATION_PERIODS, 379
- PYXX_READ_PAYROLL_RESULT, 377
- remote-enabled, 474
- REUSE_ALV_GRID_DISPLAY, 412
- REUSE_ALV_LIST_DISPLAY, 413
- REUSE_ALV_VARIANT_F4, 407
- RH_DIR_ORG_STRUC_GET, 181
- RH_GET_MANAGER_ASSIGNMENT, 583
- RH_GET_ORG_ASSIGNMENT, 584
- RH_INSERT_INFTY, 600, 601
- RH_PM_GET_STRUCTURE, 181
- RH_READ_HRPnnnn, 183
- RH_READ_INFTY, 368
- RH_STRUC_GET, 624
- RP-EDIT-NAME, 352
- SPBT_INITIALIZE, 661
- SXC_EXIT_CHECK_ACTIVE, 204

G

General authorization, 584
- profiles, 584
General structural authorization profiles, 585
General Table Display, 284, 285, 286
Generated schemas, 370
Generate Payroll infotype, 382
Generation of Quota entitlement, 388
Generation Rules for Quota Type Selection, 197
Global employment, 623
Graphical user interface (GUI), 254
Gross payroll, 52

H

HCM add-on solution, 676
HCM Processes and Forms, 607, 608, 619, 630
Hierarchical structure, 582
HR
- applicants, 571
- clusters, 572
Index

HR (Cont.)
customer-specific authorization object with context, 585
extended check with context, 569, 585
master data, 567
master data customer-specific object, 570
master data extended check, 569
master data personnel number check, 571
master data with context, 568, 585
reporting, 573
transaction code, 567
HR Administrative Services, 74, 606, 607, 610
HR Authorization Main Switches, 586
HR core, 674
HR form
address format, 457
address selection, 458, 459
assigned InfoNet, 433
attributes, 432
copying, 434, 435
enhancements, 464, 465
feature OCHRF, 461
feature PBCHO, 460
integration with ESS, 463, 464
integration with payroll, 459, 460, 461
layout editor, 433
print program, 433, 461
selection screen, 434
sort order, 462
status, 433, 435
structure, 433
Transaction PC00_M10_HRF, 462
wage type placement, 456
HR Forms Workplace, 417, 418, 430, 466, 652
form definition status, 431
program status, 431
structure status, 431
HR MetaData Workplace, 418, 466
HR Renewal, 638
HTML, 606
HTTP, 634, 641, 646
activation, 641
change, 643
logon configuration, 644
redirect URL, 644
test, 643
Human Resources Information System (HIS),
280, 325, 327, 329, 675
access object, 326
area, 328
assign user, 329
data retrieval, 328
data view, 327
graphical display, 326
task function, 329
view, 325
Hybrid design, 675
Hybrid landscapes, 674
Hybrid models, 675

IDoc, 482, 496, 499
composition, 499
display, 501
HROT_USM, 496
message type, 500, 501
iFlows, 675, 695, 696
InfoType
5046 (Collaboration Room Template), 74
Implementation, 688
HRHAP00_PMP_PM_POT, 72
InfoDimensions, 432
InfoFigures, 432
InfoNet, 431
InfoSet, 280, 286, 287, 288, 289, 290, 291, 292, 294, 295, 297, 298, 299, 300, 301, 303, 304, 311
additional fields, 289, 292, 293, 294
field group, 287, 289, 292, 294, 305
query area, 288
query generator, 294, 295, 296
/SAPQUERY/HR_ADM, 657
standard area, 288
switches, 295, 296, 297
technical name, 305
user group, 290, 291, 292, 304
InfoSet Query, 280, 294, 303, 304, 305
field group, 313
hit list, 306, 308
Index

InfoSet Query (Cont.)
output preview, 306, 307
ranked list, 307
set operations, 304, 306, 308, 310
start report, 304, 307
transfer set, 310
InfoStars, 432
Infotype, 35, 41, 78, 560, 568, 570, 571, 572, 576, 584
0000 (Actions), 36, 136, 223, 312, 359, 547, 552, 553, 578
0001 (Organizational Assignment), 36, 136, 215, 289, 312, 359, 517, 518, 519, 547, 552, 553, 560, 561, 563, 568, 569, 570, 578, 579, 684, 685
0002 (Personal Data), 36, 289, 292, 312, 547, 552, 560, 574, 578
0003 (Payroll Status), 36, 136, 171, 236, 498, 578
0005 (Leave Entitlement), 120
0006 (Address), 36, 237, 289, 300, 473, 477, 478, 484, 487, 552, 666
0007 (Planned Working Time), 36, 120, 136, 186, 385, 553
0008 (Basic Pay), 36, 53, 136, 352, 552, 553, 571, 574, 575, 585, 593, 689, 693
0009 (Bank Details), 36, 53, 136, 306, 308, 421, 423, 552
0014 (Recurring Payments/Deductions), 36, 53, 136, 312, 314, 655
0015 (Additional Payments), 36, 53, 136, 689
0016 (Contract Elements), 81
0019 (Monitoring of Tasks), 36, 236
0021 (Family Members), 81, 237
0027 (Cost Distribution), 136
0031 (Reference Personnel Number), 81, 237, 578
0040 (Objects on Loan), 427, 455
0041 (Date Specifications), 236, 352, 353
0050 (Planned Working Time), 120
0080 (Maternity Protection), 120
0081 (Military Service), 120
0082 (Additional Absence Data), 120
0083 (Leave Entitlement Compensation), 120

Infotype (Cont.)
0105 (Communication), 571, 584, 598, 621, 629, 687
0130 (Test Procedures), 120, 237, 578, 579
0167 (Health Plans), 59, 136, 553
0168 (Insurance Plans), 136
0169 (Savings Plans [401k]), 136
0170 (Flexible Spending Accounts), 136
0171 (General Benefits Information for Benefits Administration), 68, 553
0194 (Garnishment Documents US), 136
0195 (Garnishment Order US), 136
0207 (Residence Tax Area [US]), 53, 136
0208 (Work Tax Area), 53, 136, 271
0209 (Unemployment State [US]), 53, 136
0210 (Withholding Info W4/W5 [US]), 53, 136
0221 (Payroll Results Adjustment), 136
0234 (Additional Withholding Info US), 136
0235 (Other Taxes US), 136
0302 (Additional actions), 223
0315 (Time Sheet Defaults [CATS]), 120
0316, 598
0328, 598
0402 (Payroll Results), 381
0403, 381
0415 (Export Status), 498
0416 (Time Quota compensation), 121, 217
0446, 381
0457, 381
0458 (Monthly Cumulations), 381
0459 (Quarterly Cumulations), 381
0460 (Annual Cumulations), 381
0613 (Absence Donation Administration), 121
0655 (ESS Settings Remuneration Statement), 463
0672 (FMLA Event), 121
0696 (Shared Leave), 121
0709 (Person ID), 156, 348
0712 (Main Personnel Assignment), 686
0758 (Compensation Program), 68
0759 (Compensation Process), 68
0760 (Eligibility Override), 69
Index

Infotype (Cont.)

0761 (LTI Granting), 69
0762 (LTI Exercising), 69
0763 (LTI Participant Data), 69
1000 (Objects), 42, 252
1001 (Relationships), 42, 63, 252, 368
1002 (Descriptions), 42, 112
1003 (Department/Staff), 42
1005 (Planned Compensation), 42, 108
1007 (Vacancy), 545
1008 (Account Assignment), 42, 215, 551
1013 (Employee Group/Subgroup), 551
1026 (Business Event Info), 74
1029 (Course Type Info), 74
1035 (Schedule), 74
1037 (Billing/Allocation Info), 74
1060 (Demand), 74
1061 (Web Link), 74
2001 (Absences), 44, 120, 131, 223, 237, 578, 579, 586
2002 (Attendances), 44, 120, 131, 223, 237, 578, 579, 586
2003 (Substitutions), 44, 120
2004 (Availability), 120
2005 (Overtime), 45, 120
2006 (Absence Quotas), 45, 120, 237, 387
2007 (Attendance Quotas), 45, 120, 387
2010 (Employee Remuneration), 45, 120, 131, 578, 665
2011 (Time Events), 45, 120
2012 (Time Transfer Specification), 45, 120
2013 (Quota Corrections), 45, 120
2050 (Annual Calendar), 120
2051 (Monthly Calendar), 121
2052 (Weekly Calendar), 121
4004, 236
5003 (Course Content Static), 74
5004 (Curriculum Type Info), 74
5006 (Course Type Content), 74
5007 (Completion Specifications), 74
5008 (Completion Signatures), 74
5009 (Course Content), 74
5020 (Category Assignment), 71
5021 (Layout Definition), 71
5022 (Column/Cell Definition), 71
5023 (Column Access), 71

Infotype attributes, 79
Infotype characteristics, 235
Infotype contraction, 296
Infotype join, 296
Infotypes for OM, 42, 104, 252
Infotypes for Payroll, 136
Infotypes for Personnel Administration, 36, 87, 518
Infotypes for Performance Management, 71
Infotypes for Personnel Planning, 114, 115
Infotypes for PT, 44, 120, 212, 257, 385
Infotype-specific text, 572
In-period, 372
Integration Add-On, 677, 678
Integration Add-on 2.0, 678
Integration scenario, 676, 677
for compensation data, 689, 690
for evaluation data, 693
Integration solution, 674
Interactive Forms for Adobe, 607
Interface format, 484, 486, 488, 489, 492, 494, 499
 delivered, 494
 FADP, 494
 OTMU, 494, 496, 498, 500, 501
 OTUS, 494
 OTUT, 494
 Interface Toolbox, 482, 483, 488, 489, 514
 block, 485, 486, 487
 configuration, 484
 display TemSe, 491
 export, 483, 484, 489, 490, 492, 498, 505, 506
 file layout, 483, 485, 486, 488, 490, 493, 494
 header block, 487
 interface directory, 491
 outsourcing, 494, 495
 TemSe, 483, 490, 491
 trailer block, 488
 Internal cluster table
 BT, 377
 CRT, 358, 403, 409
 CVS, 388
 DFT, 403
 FEHLER, 388
 GRT, 403
 KNTAG, 388
 PSP, 388
 PT, 123, 388
 QTACC, 388
 QTRANS, 388
 RGDIR, 372
 RT, 155, 358, 370, 403, 408
 SALDO, 388
 SV, 358
 VS, 388
 WPBP, 388
 ZL, 128, 388
 Internal infotypes tables, 363
 Internal structure
 PN, 360
 Internal table
 ALL_PERNRS, 156
 Internal table (Cont.)
 ASSIGNING, 664, 665, 666
 P0001, 360
 P0007, 360
 PAYROLL-INTER-RT, 358
 PC, 366
 Internal table structure
 ZL, 59
 Internet Communication Manager (ICM), 629, 634, 641, 646
 Internet Transaction Server, 606
 Interpretation of assigned authorization, 571
 ISO code, 269
 iXML, 513

 J

 J2EE, 75
 Job, 42, 101, 367, 579, 580, 584
 Job board, 63
 Job catalog, 544
 consolidation, 544
 Job pricing, 67
 Join for multiple infotypes, 362

 K

 Key fields, 82

 L

 Layout structure, 388
 Learning, 674
 Legacy System Migration Workbench (LSMW), 515, 533, 549, 552, 557, 659
 administration, 541
 assign file to structure, 540
 assign object recording, 536
 convert data, 540
 copy project, 542
 create batch input sessions, 540
 create object, 533
 create project, 533
Legacy System Migration Workbench (LSMW) (Cont.)
create recording, 534
create subproject, 533
export project, 541, 542
field mapping, 538
file names, 536
fixed values, 539
imported data, 540
import project, 541, 543
object, 536
object attributes, 536
read data, 540
recording defaults, 535
repeat recording, 534
source fields, 537
source fields copy, 537
source structure, 537
specify files, 539
structure relations, 538
transaction recording, 534
translations, 539
user-defined routines, 539
List data, 388
Lock indicator, 578
Logged changes in infotype data, 338, 339, 340
delete, 340
long-term document, 339
short-term document, 339
Logical database, 147, 343, 589, 598
GET PERNR, 660
PAP, 147, 344, 573
parallel processing technique, 660
PCH, 147, 158, 172, 181, 344, 366, 654
PNP, 135, 147, 150, 152, 157, 295, 343, 344, 358, 573, 574, 654, 666
PNPCE, 135, 147, 156, 157, 287, 289, 295, 300, 312, 316, 343, 344, 358, 433, 455, 573, 654
Logical system, 502, 503
define, 502
FILEPORT, 495, 501
Log of Report Starts, 336, 338
Long-term incentives, 68
Loops, 361
LUW, 521, 523
Macro, 164, 345
BREAK, 164
PNP_SW_SKIP_PERNR, 589
PROVIDE_FROM_FIRST, 363
RH-GET-TBDATE, 173
RH-GET-TBDATE, 369
RH-SEL-KEYDATE, 173, 369
RH-SEL-ONE_OBJID, 369
RH-SEL-ONE-OBJID, 172
RP-EDIT-NAME, 166
RP-EXP-C2-CU, 138
RP-EXP-C2-RU, 138
RP-IMP-C2-CU, 138
RP-IMP-C2-RU, 138, 139, 379
RP-IMP-C4-LA, 185
RP-IMP-C4-LB, 185
RP-IMP-C4-SA, 185
RP-IMP-C4-SB, 185
RP-LOWDATE-HIGHDATE, 166
RP_PROVIDE_FROM_FRST, 165, 168
RP_PROVIDE_FROM_LAST, 168, 363
RP_READ_ALL_TIME_ITY, 170, 385
RP_READ_INFOTYPE, 169
RP_SEL-EIN-AUS-INIT, 166
RP_SET_DATA_INTERVAL, 167, 360, 590
RP_SET-NAME-FORMAT, 166, 352
RP_UPDATE, 171
TRMAC, 184, 185
Main authorization switches, 585, 590
Maintain text, 237
Maintenance Optimizer, 610
Management of Global Employees (MGE), 147, 156, 686
Manager Add-On, 607, 626, 627, 628
Manager Self-Service (MSS), 66, 584, 598, 605, 606, 612, 615, 626, 645
application configuration, 635, 636
authorizations, 628
business function, 628
business package, 626, 627
component configuration, 637
configuration, 634
customization, 638
data tracking, 640
Manager Self-Service (MSS) (Cont.)
  documentation, 628
  integration, 627
  launchpad, 632, 634, 639
  navigation, 629
  Web Dynpro for ABAP, 626, 628
Mapping, 695
Messages, 388
MetaData, 418
MetaDimension, 418, 420, 421 427  
  create, 427
  data selection, 423, 429
  read with MetaStar, 423, 427
  selection, 422
MetaField, 422, 423
  attribute, 422, 427
  create, 428
  key, 422, 427
MetaFigure, 420, 426
  attribute, 426
MetaNet, 418
  composition, 420
  country grouping, 418, 419
  creating, 419
  customization, 426, 429
  inheritance, 419
MetaNets
  maintain, 420
MetaStar, 418, 420, 421, 424
  assign to MetaNet, 429
  composition, 424
  create, 428, 429
  data selection, 424, 426
Method, 164, 184, 230, 591
  APPROVE_CATS_CLASSIC, 206
  CHECK_AUTHORITY_VIEW, 592
  CHECK_AUTHORIZATION, 591
  CHECK_AUTH_PLAN1, 592
  CHECK_MAX_INFTY_AUTHORIZATION, 591
  CHECK_MAX_LEVEL_AUTHORIZATION, 591
  CHECK_MAX_PERNR_AUTHORIZATION, 591
  CHECK_MAX_PERNR_AUTHORIZATION, 591
  CHECK_MAX_PERNR_AUTHORIZATION, 591
  CHECK_MAX_SUBTY_AUTHORIZATION, 591
Method (Cont.)
  CHECK_MIN_INFTY_AUTHORIZATION, 591
  CHECK_MIN_LEVEL_AUTHORIZATION, 591
  CHECK_MIN_PERNR_AUTHORIZATION, 591
  CHECK_MIN_PERNR_AUTHORIZATION, 591
  CHECK_MIN_SUBTY_AUTHORIZATION, 591
  CHECK_PERNR_AUTHORIZATION, 591
  DELAYED_CONSTRUCTOR, 591
  FILL_DATE_VIEW, 592
  FILL_HYPER_VIEW, 592
  GET_EMAIL, 687
  GET_MANAGER, 684
  IF_EX_CATS_REPORTING~APPROVE_ 
    CATS_CLASSIC, 210
  IF_EX_HRPAD00AUTH_TIME~CONSIDER_ 
    _SY_DATUM_EXIT, 593
  IN_UPDATE, 243
  MAP_INT_NAME_TO_PROXY_NAME, 689
  MAP_INT_VALUE_TO_PROXY_VALUE, 689
  SET_ORG_ASSIGNMENT, 591
  SET_PARTIAL_ORG_ASSIGNMENT, 591
  UPDATE_DB, 243
Methods, 389
Microsoft Sharepoint InfoPath forms, 75
Middleware, 677, 678, 679, 682, 689, 696
MiniApps, 564
Mobile HCM, 674
Model, 95
Modification Assistant, 216
Modification Browser, 216
Modifications, 216
Module pool, 78, 92, 253
MOLGA, 209, 229, 379, 384, 561, 563
  Molga 99, 383
MS Excel, 388

N

Net payroll, 52
NetWeaver Business Client, 74, 605, 615, 617
  desktop, 615, 616
Index

NetWeaver Business Client (Cont.)
  HTML, 615, 618, 620, 621, 629, 641, 645
  work centers, 617
Next operations, 187

O

Object, 252, 584
  Business Partner, 63
  Candidate, 64
  Central Person, 63
Object and Data Provider, 630, 631, 640
  Data Provider, 630
  Object Provider, 630
  Organizational view, 630
Object-Based Navigation, 632, 634, 645
Object class, 570
Object ID, 113, 366, 582
Object-oriented (OO), 164
  programming model, 95
Object plug-in, 212
Objects, 581
Object status, 367
Object type, 41, 42, 100, 348, 366, 576, 580, 582
Old result, 374
OM module, 563, 567, 576, 579
OM table infotypes, 111
On-premise, 61, 673, 675, 697
Operation, 265
  ADDWT, 60
  BREAK, 276
  OUTWP, 276
Organizational assignment, 570, 573, 575, 584, 589
Organizational Chart Visualization, 638, 639
Organizational data, 677, 689
Organizational identifier, 579
Organizational key, 559, 560, 561, 562, 563, 568, 572, 574, 579
Organizational key control, 561
Organizational plan, 580
Organizational structure, 580, 581, 582, 584, 585
Organizational unit, 38, 42, 101, 182, 367, 563, 579, 580, 585, 589
  lowest supervisory level, 544
OSS Notes, 193
Output controller, 596

P

P0001-WERKS, 362
Pair table, 123
Parallel and Distributed Reporting Scheduler, 652, 654
  background processing, 653
Parallel Evaluation Scheduler, 651
Parallel Payroll Scheduler, 648, 651
Parallel processing, 648, 653, 671
Payment medium programs, 575
Payroll accounting results/international, 370
Payroll Administrator, 569
Payroll area, 38, 57, 563
Payroll cluster table, 369
Payroll control record, 57, 135
Payroll data, 677
Payroll Drive Program infotypes, 53
Payroll Evaluation infotypes, 381
Payroll form, 380
Payroll frequency, 54
Payroll infotypes, 370
Payroll (PY) module, 52
Payroll results, 356, 369, 373, 377, 379
  Payroll results directory table, 374
  Payroll results infotypes, 381
  Payroll results tables, 140
PC-BEGDA, 366
PC-ENDDA, 366
Performance and goals, 674
Performance Management, 70
Period, 583
Period of responsibility, 588, 589
Permissibility indicator, 96
Person, 101
Person ID, 156, 348
Personnel action, 37
Personnel Administration module, 35, 582
Personnel area, 560, 568, 572, 579
Personnel assignments, 687
Personnel calculation rule, 49, 59, 265
   TP10, 268
  X015, 60
Personnel development, 576
Personnel number (PERNR), 39
Personnel officer responsible, 572
Personnel Planning, 576
Personnel structure, 38
Personnel subarea, 563, 684
Personnel subarea grouping, 197, 201
Personnel work schedule, 388
Person selection period, 153, 345, 359
Planned working time, 45
Planning status, 41
Plan version, 366, 576, 581
PN-BEGDA, 360
PN-ENDDA, 360
Portal Runtime Environment, 75
Position, 42, 101, 367, 563, 579, 580, 589
Posting, 63
Previous result, 374
Print parameters, 597
Process after input (PAI), 92, 225, 227, 230, 242, 518
Process before output (PBO), 92, 225, 227, 230, 242, 518
Processing mode, 576
Profile, 563, 583, 596
Profile authorization groups, 598
Profile authorizations for CATS, 597
Profile Generator, 563, 564, 565, 566, 567, 579
Program
   DBPNPCECOM, 345
   DBPNPCCOM, 345
   EXAMPLE_PNP_GET_PAYROLL, 155, 379
   H99_HRFORMS_CALL, 652
   HMXCALCO, 59
   HRSFI_D_COMP_EXT, 691
   HRSFI_MONITORING_COMP, 693
   HRSFI_MONITORING_EMPL, 682
   MP000100, 518, 519
   MP000200, 560
   MP000600, 223
   MP041640, 218
   MP200000, 261
   MP900200, 239
Program (Cont.)
   MPPAUTCON, 588
   MPPAUTZZ, 570, 588, 595
   MPPERS00, 227, 230, 246
   PCBURZnn0, 269
   PCBURZnn0_CE, 269
   PCFDCZnn0, 269
   PCFDCZnn0_CE, 269
   performance, 647
   RHALTD00, 516, 549, 550
   RHBAUS00, 588
   RHBAUS01, 588
   RHBAUS02, 588
   RHCHECK0, 115
   RHGRIN23, 328
   RHINTEO0, 553
   RHINTE20, 552, 553
   RHKOST00, 103
   RHPREL00, 102
   RHPROFL0, 588
   RHRHDL00, 552
   RH_SFI_ACTIVATE_COMP_DATA, 690
   RH_SFI_CLEANUP_COMP_IMP, 690
   RH_SFI_CLEANUP_COMP_REPL, 690, 691
   RH_SFI_IMPORT_COMP_DATA, 690
   RH_SFI_SYNCH_COMP_DATA, 690, 691
   RH_SFI_SYNCHRONIZE_EMPL_DATA,
   679, 682
   RH_SFI_TRIGGER_EMPL_DATA_REPL, 679
   RHSTRU00, 158, 366
   RHT777DCHECK, 116
   RHUSERRELATIONS, 588
   RPABRI00, 332, 382
   RPC2RUU0, 380
   RPC2RX09, 380
   RPCALCn0, 128, 265
   RPCALCnX, 277
   RPCALCU0, 59, 135, 138, 265, 269, 555, 648
   RPCALCxx, 158, 356, 578
   RPCEDTU0, 380
   RPCEDTU0, 380
   RPCLSTB1, 126, 387
   RPCLSTB2, 129, 387
   RPCLSTRU, 380
   RPCS0000, 651
   RPSCSC000, 158, 648
   RP_DIST_REP_EXAMPLE, 653
Index

Program (Cont.)

- RPLMIT00, 148
- RPMREP00, 653
- RPMUST01, 380
- RPMUST02, 380
- RPTBPC10, 218
- RPTEZL00, 129
- RPTIME00, 46, 49, 121, 158, 197, 198, 265, 277, 387, 574, 575, 578
- RPTMO200, 269, 273
- RPTQTA00, 46
- RPUACG00, 570, 588, 595
- RPUAUD00, 185, 364, 575
- RPUAUDDL, 340
- RPUDELPN, 695
- RPUDELP, 695
- RPUKTO00, 516, 554
- RPUKMDO00, 191
- RPUPROTU, 338
- RPUSTD00, 516, 552
- SAPDBPAP, 589
- SAPDBPNP, 147, 166, 589
- SAPFP51T, 269
- variant, 651
- Program operations, 187
- Projection, 361, 362

Q

- Qualification, 101
- QuickViewer, 280, 297, 301, 302
  - data source, 299
  - export to SAP Query, 302, 303
  - layout, 299, 300, 301, 302
- Quota accruals, 50
- Quota generation rules, 50

R

- R/2, 87, 242
- Rapid Deployment Solutions (RDS), 695
  - packages, 696
- Read payroll results, 376
- Receiver adapter, 695

- Receiver components, 695
- RECEIVE RESULTS FROM FUNCTION, 663
- Recruiting, 674, 677
- Recursion check, 367
- Reference table, 283
- Relationships, 41, 100, 252, 368, 580, 582
- Relationship types, 103
- Repetitive field structures, 352
- Replication of employee data, 684, 685, 686
- Report category, 149, 345, 346, 348
- Reporting
  - audit, 336, 337, 338, 339
  - payroll infotypes, 330, 331, 332
- Reporting period, 367
- REPORT TYPE, 117
- Report variant, 682
- Repository Browser, 241
- Repository Information System, 508
- Requisition, 63
- Retroactive Accounting Payroll, 238
- Retro-calculation, 137, 372
- Return operations, 187
- RFC, 481, 482, 652, 653, 671
  - asynchronous, 482
  - background, 482
  - connection types, 482
  - synchronous, 482
  - transactional, 482, 504
- RMAC macros, 164, 345
- Role, 559, 563, 564
- Role assignment, 564
- Role maintenance, 585
- Root object, 366, 368, 581, 582
- Rule Editor, 267

S

- Sample reports, 380
- SAP add-on country versions, 384
- SAP batch job, 682
- SAP BusinessObjects, 625
- SAP Data Dictionary (DDIC), 224, 370
- SAP CRM, 607
- SAP E-Recruiting, 605
- SAP ERP 6.0, 677
SAP ERP HCM, 674, 689
  data structures, 77
  solution portfolio, 674
SAP Floorplan Manager, 638
  FPM_OIF_COMPONENT, 638
SAP GUI, 615, 616, 617
SAP HANA Cloud Integration, 676
SAP Hybrid Deployment, 676
SAP NetWeaver Business Client (NWBC), 63
SAP NetWeaver Portal, 74, 75, 606, 610, 615, 618, 620, 626
SAP NetWeaver Process Integration, 675, 677, 678, 689, 695, 696
SAP object types, 101
SAP Plant Maintenance (PM), 129
SAP Process Integration, 509
SAP Project System (PS), 129
SAP Query, 280, 294, 297, 302, 310, 318, 657
  basic list, 314
  basic list header, 315
  change lock, 312
  control levels, 314
  copy query, 321
  delete query, 321
  field output, 314
  generated program, 320, 321
  Graphical Query Painter, 312
  graphics, 315
  list line output, 314, 317
  multi-line output, 317, 318
  query area, 302, 311
  report-to-report, 319
  retrieve query, 322
  selection screen, 313
  test query, 315, 318
  text translation, 323, 324
  Transport Tool, 322
  user group, 311
SAP Talent Visualization (STVN) by Nakisa, 65
SAP UI 5, 617
Schema, 49, 59, 265
  TM000, 267
  U000, 59
  ULK9, 555
Schema Editor, 267
Screen header, 237
Screens, 78, 253
Screen structure, 97, 243
Search term, 367
Security troubleshooting, 602
Selection period, 359
Selection screen, 366
  Selection screen layout, 389
Select options, 347
Sender, 695
Sender adapter, 695
Sequential evaluation, 159, 160, 366, 367
Service interface, 688
Set breakpoints, 276
Set structure conditions, 367
SFPI, 431
SGML, 505
Smart Forms, 417, 652
SMTP, 641
SOAP, 508
Social Jam, 673
Social Talent Management, 674
Software Change Registration (SSCR), 216
Source code plug-in, 212
Spool authorization, 596
Spool authorization group, 596
SQL, 285
  ABAP, 285
SSFO, 431
Status indicator, 379
Status overlap, 367
Status vector, 367, 582
Structural authorization, 563, 579, 580, 592, 598
Structural authorization check, 582, 584, 589
Structural authorization profile, 585, 588, 589
Structural evaluation, 159, 160, 366, 368
Structural profile, 580, 583, 584, 593
Structure, 78, 238, 253
  BDCDATA, 519, 521, 523, 534
  BDCSMGCOLL, 523
  CI_nnnn, 86, 115
  CI_P0006, 225
  E1P1000, 549, 551
  E1P1001, 551
  GROUP, 157
  HCMT_BSP_PA_yy_Rnnnn, 91, 99
Index

Structure (Cont.)
  HCMT_BSP_PA_yy_Rnnnn_LIN_z, 99
  HRI1001, 113
  HRI1005, 108
  HRIDAMIN, 108
  HRIKEY, 106, 107
  HRIKEYL, 107
  HRInnnn, 108
  HRSFI_S_TRANSFER_DATA, 688
  HRSFI_S_TRANSFER_DATA_LABEL, 689
  HRTnnnn, 112
  I001P, 230
  I503, 230
  P2001, 170
  P9002, 239
  PAKEY, 82
  PAnnnn, 82
  PAYROLL, 154, 356
  PAYUS_RESULT, 140
  PC261, 139, 372
  PDPS, 385
  PERNR, 359
  PERSON, 156
  Pnnnn, 86, 110
  PS0002, 84
  PS9002, 233
  PSHD1, 82
  PSnnnn, 84
  PSYST, 230, 244
  PT1002, 112
  Pttnnn, 112
  PYORGSCREEN, 154, 358
  PYTIMESCREEN, 154, 358
  ZHCMT_BSP_PA_yy_Rnnn, 99
  ZHCMT_BSP_PA_yy_Rnnnn_LIN_z, 99

Subfeature, 187

Sub-schema
  UTR0, 60

Subtype, 79, 560, 568, 570, 571, 572, 576, 584

SuccessFactors, 673, 675

SuccessFactors Business Execution (BizX), 673, 674, 677, 678, 688, 689
  integration, 696

SuccessFactors Recruiting, 677

SuccessFactors Transfer Monitoring, 682

Succession and development, 674

Super class, 240

Support packages, 608

Switch Framework, 611, 612, 615, 626

SY-DATUM, 593

Symmetrical double verification, 573

Symmetrical verification process, 576

Sync employee data with SuccessFactors, 679

T
Table

  CATSDB, 129
  CATSHR, 131
  foreign key, 283, 301
  HRAUTH_DIST_REP, 654
  HRHAP, 71
  HRHAP_ACT_LOG, 71
  HRHAP_ANON, 71
  HRHAP_APP, 71
  HRHAP_APPER, 71
  HRHAP_ARC_GRST, 71
  HRHAP_BASIC, 71
  HRHAP_FINAL, 71
  HRHAP_FURTHER, 71
  HRHAP_OBJECT, 71
  HRHAP_OFFLINE, 71
  HRHAP_OTHERS, 71
  HRHAP_P, 71
  HRHAP_P_APPER, 72
  HRHAP_PART, 72
  HRHAP_P_D, 72
  HRHAP_P_E, 72
  HRHAP_PROCESS, 72
  HRHAP_P_T, 72
  HRHAP_SEM_OBJ, 72
  HRHAP_T, 72
  HRHAP_TAB, 72
  HRHAP_TAB_DATA, 72
  HRHAP_TAB_T, 72
  HRP1000, 113, 162
  HRP1001, 113
  HRP1005, 110
  HRP1008, 163
  HRPnnnn, 110, 112
  HRPY_RGD, 357
  HRPY_WBPB, 357
<table>
<thead>
<tr>
<th>Table (Cont.)</th>
<th>Table (Cont.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HRSFI_D_COMP_ERR, 691, 692, 695</td>
<td>T549C, 190</td>
</tr>
<tr>
<td>HRSFI_D_COMP_EXT, 691, 695</td>
<td>T549D, 190</td>
</tr>
<tr>
<td>HRSFI_D_COMP_IMP, 691, 695</td>
<td>T552A, 385</td>
</tr>
<tr>
<td>HRSFI_D_COMP_STA, 692, 695</td>
<td>T555E, 272</td>
</tr>
<tr>
<td>HRSFI_D_COMP_STAT, 691</td>
<td>T558B, 554</td>
</tr>
<tr>
<td>HRSFI_D_ERR_LOG, 679, 680</td>
<td>T558C, 555</td>
</tr>
<tr>
<td>HRSFI_D_EXTR_LOG, 679, 680, 688, 695</td>
<td>T582A, 78, 80, 185, 235, 517</td>
</tr>
<tr>
<td>HRT1002, 112</td>
<td>T582C, 225</td>
</tr>
<tr>
<td>join, 288, 297, 301, 303</td>
<td>T582ITD, 96</td>
</tr>
<tr>
<td>NT1, 123</td>
<td>T582ITVCLS, 95, 96, 99, 240</td>
</tr>
<tr>
<td>PA0002, 85, 162</td>
<td>T582UICONVCLS, 240</td>
</tr>
<tr>
<td>PA2002, 133</td>
<td>T584A, 578</td>
</tr>
<tr>
<td>PA9002, 239</td>
<td>T585A, 185, 364, 548</td>
</tr>
<tr>
<td>PAannnn, 85</td>
<td>T585B, 185, 364, 548</td>
</tr>
<tr>
<td>PRel, 87</td>
<td>T585C, 185, 364, 548</td>
</tr>
<tr>
<td>PTEX2000, 131</td>
<td>T588C, 554</td>
</tr>
<tr>
<td>PTEX2010, 131</td>
<td>T588G, 238</td>
</tr>
<tr>
<td>PTQUODED, 121</td>
<td>T588JTVCHCK, 100</td>
</tr>
<tr>
<td>RGDIR, 138</td>
<td>T588M, 88, 99, 145, 223, 238</td>
</tr>
<tr>
<td>T5U8C, 554, 555</td>
<td>T588MFPROPC, 99</td>
</tr>
<tr>
<td>T7XXSCE_GRP, 624</td>
<td>T588UICONVCLS, 100, 243</td>
</tr>
<tr>
<td>T52RELID, 572</td>
<td>T588UICONVCLASS, 97</td>
</tr>
<tr>
<td>T77CD, 116, 255</td>
<td>T588Z, 246, 548</td>
</tr>
<tr>
<td>T77EO, 102, 113, 582</td>
<td>T591A, 80, 237</td>
</tr>
<tr>
<td>T77FC, 582</td>
<td>T596F, 648</td>
</tr>
<tr>
<td>T77ID, 114, 116, 239, 255</td>
<td>T599B, 149</td>
</tr>
<tr>
<td>T77NI, 255</td>
<td>T599R, 337</td>
</tr>
<tr>
<td>T77PR, 581, 582, 583</td>
<td>T599W, 149</td>
</tr>
<tr>
<td>T77SO, 69, 73, 95, 546, 585, 586, 589, 596</td>
<td>T777D, 92, 112, 114, 116, 238, 239, 255</td>
</tr>
<tr>
<td>T775Q, 571</td>
<td>T777ITCLASS, 118, 255</td>
</tr>
<tr>
<td>T77UA, 579, 584, 593</td>
<td>T777I, 105, 255</td>
</tr>
<tr>
<td>T500L, 137, 265, 269, 357, 370, 379</td>
<td>T777IFPROPC, 255</td>
</tr>
<tr>
<td>T500L_ADDON, 137</td>
<td>T777IFPROPS, 255</td>
</tr>
<tr>
<td>T510, 54</td>
<td>T777IT, 105, 255</td>
</tr>
<tr>
<td>T512T, 286</td>
<td>T777Z, 105, 255</td>
</tr>
<tr>
<td>T512W, 286</td>
<td>T778A, 182</td>
</tr>
<tr>
<td>T522F, 166, 352</td>
<td>T778O, 101, 256</td>
</tr>
<tr>
<td>T522N, 166</td>
<td>T778S, 582</td>
</tr>
<tr>
<td>T527, 561</td>
<td>T778T, 105, 255, 256</td>
</tr>
<tr>
<td>T527A, 561, 562, 563</td>
<td>T778U, 256</td>
</tr>
<tr>
<td>T527O, 561</td>
<td>T778V, 104</td>
</tr>
<tr>
<td>T542A, 301</td>
<td>TDCT, 255</td>
</tr>
<tr>
<td>T542T, 301</td>
<td>TEVEN, 122</td>
</tr>
<tr>
<td>T549B, 190</td>
<td>THRFORM_CLASS, 459</td>
</tr>
<tr>
<td></td>
<td>TRMAC, 164, 345</td>
</tr>
<tr>
<td></td>
<td>USOBT, 565</td>
</tr>
</tbody>
</table>
Index

Table (Cont.)

USOBX_C, 565
USOBX, 565
USOBT_C, 565
Tables in PCL1 Cluster B1, 125
Tables in PCL2 Cluster B2, 127, 128
Talent Management, 61, 673
Talent profile, 65
Talent Profile Attribute Infotypes, 66
Talent profile information, 66
Talent review meeting (TRM), 67
Task, 101
Tax Reporter, 143
Team Viewer, 627, 631
Technical attributes, 238
Technical depth, 367
Test procedures, 559, 578, 587
Texts for generated schemas, 370
Third-party remittance, 61
Time accounting results, 370
Time administrator, 574, 579
Time constraint, 81, 106, 236, 255
Time-Dependent Data, 358
Time entry profiles, 597
Time evaluation, 49
  Time evaluation data, 387
Time events, 122
Time logic, 589, 593
Time management data, 384
Time Management (PT) module, 44
Time pairs, 123, 388
Time type, 197, 201
Time Wage Types, 388
Tolerance Time for Authorization Check, 589
Training and Event Management, 576
Transaction, 565
  BAPI, 475
  CAT2, 48, 598
  CAT6, 47, 131, 578
  CATA, 47, 131, 578
  CATS_APPR_LITE, 48, 209
  CMOD, 198, 227
  FILE, 496, 550
  HRFORMS, 430
  HRFORMS_METADATA, 418
  HRSFI_COMP_DATA_ACT, 690
  HRSFI_COMP_DATA_IMP, 690
Transaction (Cont.)
  HRSFI_SYNCH_COMP_DATA, 690
  LPD_CUST, 621, 632, 639
  LSMW, 533
  NWBC, 618
  OOAC, 585, 586
  OODT, 549
  PA03, 57, 548
  PA20, 36, 92, 330, 332
  PA30, 36, 58, 92, 234, 237, 518, 522,
    534, 548, 552, 560, 562
  PA40, 37
  PA42, 659
  PA70, 655
  PA71, 659
  PB10, 62
  PC00_M10_HRF, 462
  PC00_M10_OTEM, 498, 505, 506
  PC00_M10_OTEX, 506
  PC00_M38_RPCSC000, 648
  PC00_M99_CWTR, 334
  PC00_OTTM, 506
  PE01, 59, 267
  PE02, 60, 267, 276
  PE03, 186, 244, 458, 461, 597
  PE04, 59, 60, 268, 270
  PE51, 456, 458, 459, 460
  PFCG, 564, 579, 584, 585, 601, 615, 620,
    624, 631
  PM01, 222, 225, 226, 235, 238, 241,
    257, 331
  PP01, 41, 117
  PP02, 117, 518, 550
  PP03, 117
  PPC1, 116
  PPC1, 253, 255
  PPIS, 325
  PPOM_OLD, 183
  PPOSE, 183, 552, 553
  PT60, 49
  PT_CLSTB1, 126
  PT_CLSTB2, 129
  PU12, 483, 492
  PU19, 143
  recording, 526
  SA38, 92
  S_AHR_61016380, 338
Transaction (Cont.)
_S_AHR_61016381, 337
_S_AHR_61018754, 141
SALE, 502
SE11, 82, 106, 142, 234, 241
SE16, 137, 144, 280, 372
SE16N, 284, 285
SE18, 205, 592, 594
SE19, 205, 262, 466, 594
SE24, 208, 513, 593, 594
SE35, 518
SE36, 148
SE37, 173, 356, 476, 506
SE38, 92, 155, 190, 196, 198, 213, 225, 345, 651
SE51, 225, 241
SE78, 443
SE80, 150, 241, 508
SE81, 174
SE84, 196, 207
SE95, 216
SFW5, 610, 612, 619, 628
SHDB, 526, 532
SICF, 641
SM30, 517
SM31, 234, 272
SM35, 529, 540, 550, 657, 658
SM36, 597
SM37, 649, 650, 652
SM59, 481
SMARTFORMS, 436
SMARTSTYLES, 439
SMOD, 194
SNOTE, 216
SOAMANAGER, 508, 509
SP01, 596
S_PH0_48000542, 506
S_PH0_48000543, 506
SPRO, 561
SPROXY, 688
SQ01, 311
SQ02, 286, 287, 288, 322
SQ03, 290
SQ07, 323
ST01, 602
STRANS, 512
SU20, 594

Transaction (Cont.)
SU21, 570, 585, 594, 595
SU22, 595
SU24, 565, 566
SU53, 601
SU56, 601
SWO4, 471
WE02, 500
WE20, 495, 497, 504
WE21, 496
Transaction Recorder, 526
create ABAP program, 528
create BDC session, 527
create function module, 528
create recording, 526
create test data, 529
process, 527
Transfer pool for quota, 388
Transparent database tables, 78
Transparent table, 78, 110
Troubleshooting authorization issues, 601
TXT, 678

U

UIBB, 637
Unicode, 505
Unlocked data records, 567, 658
SM36, 597
SM37, 649, 650, 652
SM59, 481
SMARTFORMS, 436
SMARTSTYLES, 439
SMOD, 194
SNOTE, 216
SOAMANAGER, 508, 509
SP01, 596
S_PH0_48000542, 506
S_PH0_48000543, 506
SPRO, 561
SPROXY, 688
SQ01, 311
SQ02, 286, 287, 288, 322
SQ03, 290
SQ07, 323
ST01, 602
STRANS, 512
SU20, 594

Validation
cross-field, 516
Variable balances, 388
Variable key, 561

V

Vacation accrual, 197
Index

View, 95
  V_T7XSSCE_GRP, 623
  V_T7XSSCESTATFLT, 625
  V_T555E, 272
  V_T582A, 136
  V_T582ITVCLAS, 243
  V_TWPC_ORGVWG_P, 631

W

Wage type, 54
  evaluation class, 456, 457
Wage Type Reporter, 61, 333, 334, 336, 555
    archived results, 335
    error log, 336
    object selection, 335
    output display, 336
    payroll interval, 334
    period comparison, 335
    period determination, 335
    selection, 334
Wage types, 266
Web Dynpro Application, 693
Web Dynpro for ABAP, 73, 74, 606, 611, 612, 615, 632, 645
    application configuration, 634
    configuration, 634, 635
Web Dynpro for ABAP (Cont.)
    configuration editor, 636
Web Dynpro for Java, 75
Web service, 508
    definition, 507, 508
    testing, 508
Web services, 505, 506
    create, 506
Work center, 357, 101
Workforce analytics, 673, 674, 677, 693
Workforce planning, 673, 674, 693
Work schedule rules, 385
WSDL, 506

X

XML, 504, 508, 510, 514
  ABAP development, 510
  deserialization, 512
  IF_IXML definitions, 513
  serialization, 511, 512
  transformation, 511, 512
  transformation simple, 512, 513
XSLT, 514
XSLT Editor, 512, 513
XSS services, 243