



Modifikationen im HCM

Möglichkeiten und Vorgehensweisen bei der
Umsetzung eines Prozesses in der Personalwirtschaft





Ort der Umsetzung immer abhängig vom Coding des SAP-Programms

- | | |
|--------------------------|-------------------------------------|
| ≡ Customizing | >> Idealfall (keine Progr.)= k.Mod. |
| ≡ Customer-Exit | >> falls vorhanden, geringe Progr. |
| ≡ BAdI | >> falls vorhanden, geringe Progr. |
| ≡ Modifikationsassistent | >> Notfall, da umfangreiche Progr. |



Suche nach Customer-Exit/BAdI

- Schritt 1: Suche im SAPNET nach geeignetem Hinweis
- Schritt 2: Suche im Customizing-Leitfaden
- Schritt 3: Suche im Coding des jeweiligen Reports

Geeignete Suchbegriffe: cust, exit oder badi



Beispiel: Customizing

- Problem: Bei einer Änderung des Geburtsdatums soll ein zusätzlicher Infotyp Terminverfolgung eingefügt werden.

Einführungsleitfaden anzeigen

Existierende BC-Sets BC-Sets zur Aktivität Akt

Struktur

- Personalmanagement
 - Globale Einstellungen im Personalmanagement
 - Organisationsmanagement
 - Personalentwicklung
 - Personaladministration
 - Grundeinstellungen
 - Persönliche Daten
 - Organisatorische Daten
 - Vertragliche und betriebliche Vereinbarungen
 - Abrechnungsdaten
 - Auswertungsgrundlagen
 - Personalstammblatt Schweiz
 - Mitarbeiterqualifikation
 - Kommunikation
 - Anpassung der Arbeitsabläufe
 - Informationstypen
 - Infotypmenüs
 - Maßnahmen
 - Dynamische Maßnahmen
 - Mail-Anschluß für Infotypänderung einrichten



Keine Modifikation nötig !!

Sicht "Dynamische Maßnahmen" ändern: Übersicht

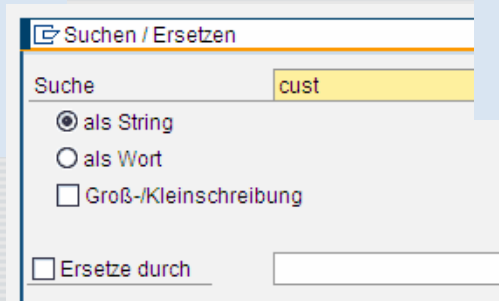
Neue Einträge

Infy	Art	Feldnam	FC	Nr	A	Variabler Funktionsteil
0002		GBDAT	06	35	I	INS, 0019, (F_TMART), , (F_NBDAT), (F_NBDAT)
0002		GBDAT	06	40	W	P0019-TMART=F_TMART
0002		GBDAT	06	45	W	P0019-TERMN=F_NBDAT



Beispiel für Customer-Exit 1

- Problem: Eingabeprüfung im Standardinfotyp erforderlich
- Suche im SAPNET und Customizing-Leitfaden erfolglos
- Suche im Eingabeprogramm des Infotypen MP000100 (IT0001)



```
CALL CUSTOMER-FUNCTION '002'
  EXPORTING
    TCLAS = PPAR-TCLAS
    INNNN = CPREL
    PSAVE = CSAVE
    IPSYST = PSYST
    I001P = T001P
    I503 = T503
  IMPORTING
    INNNN = CPREL
    SHOW_DATA_AGAIN = SHOW_DATA_AGAIN
  CHANGING
    IPREF = PREF
  EXCEPTIONS
    ERROR_OCCURRED = 1
    OTHERS = 2.
```



Beispiel für Customer-Exit 2

Transaktion CMOD zur Aktivierung

Attribute Erweiterungsprojekt ZTEST

Zuordnung Erweiterung Komponenten

Projekt ZTEST

Kurztext

Repository Infosystem: Exits suchen (200 Treffer)

Exitname	Kurzbeschreibung
P01ABAV8	Verarbeitung von Rentenbasisbezügen
P01ABAV9	Abrechnungsfunktionalität Gehaltsumwandlung
PA30CAR1	
PA30CAR2	
PA30CRM1	Berechnung des Mitarbeiter-Eigenbeitrags Firmenwagen (DK
PACKING	Erweiterung für Änderungen im Verpacken
PACKINST	Erweiterungen für Packvorschriften
PACKMOD1	Userexit für kundenspez. Funktionen auf Versandhilfsmitt
PADR0001	PA: Ableitung Finanzobjekt aus Geschäft
PARA0001	Erweiterung zur Bestimmung der Anzahl Perioden
PARA0002	Erweiterung zum Berechnen des Jahresgehalts; IT0008,14,1
PATS0001	Kundenerweiterung zu FB HR_AT_GET_ORG_DATA
PB020001	HR-CH: Customer-Exit zur Rentenalterbestimmung
PB160001	
PB160002	
PB160003	
PBAS0001	PA: Personaladmin./Beschaffung: Vorschlagswerte u. Prüfu
PBAS0002	PA: Erweiterungen für Personaladministration - Menü
PBASRP01	PA: Userexit verteiltes Reporting
PBEN0001	Kundenerweiterung für Merkmal BAREA (ArbgLeistungsbereic
PBEN0002	Kundenerweiterung für Merkmal BAREA (ArbgLeistungsbereic

SAP-Erweiterungen in Erweiterungsprojekt ZTEST

Erweiterung Komponenten

Repository Infosystem: Exits suchen

Standard Selektionen

Exitname p-1

Kurztext

Paket

Anwendungskomponente

Einstellungen

Max. Trefferzahl 200



Beispiel für Customer-Exit 3

Beispiel einer Umsetzung bei Unicode (alle Infotypen laufen über eine Routine)

data: xxxx0001 like p0001.

```
class cl_hr_pnnnn_type_cast definition load.
```

```
case innnn-infty.
```

```
  when '0001'.
```

```
    call method cl_hr_pnnnn_type_cast=>prelp_to_pnnnn
```

```
      exporting
```

```
        prelp = innnn
```

```
      importing
```

```
        pnnnn = xxxx0001.
```

```
IF XXXX0001-MSTBR NE SPACE.
```

```
  SELECT      * FROM ZZHRMSTBR
```

```
    WHERE MSTBR      = XXXX0001-MSTBR.
```

```
ENDSELECT.
```

```
IF SY-SUBRC NE 0.
```

```
  MESSAGE ID '38' TYPE 'E' NUMBER '016'
```

```
    WITH 'Meisterbereich nicht korrekt.'.
```

```
ENDIF.
```

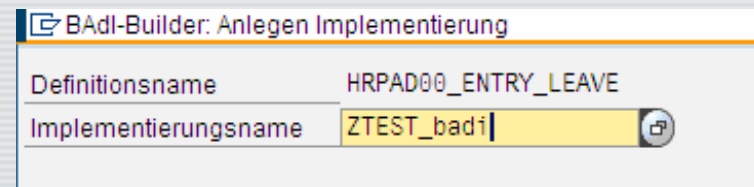
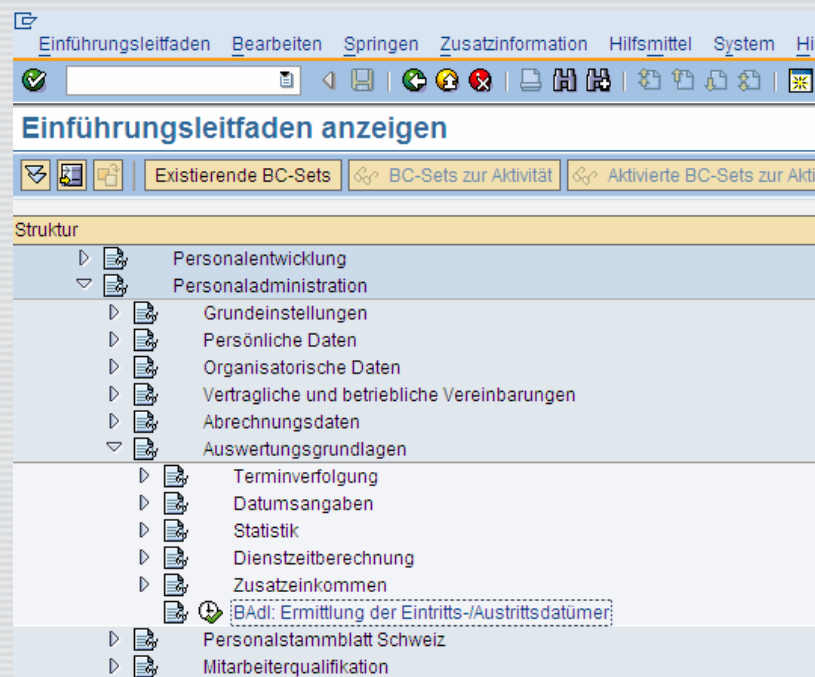
```
ENDIF.
```

```
endcase.
```



Beispiel für BAdI 1

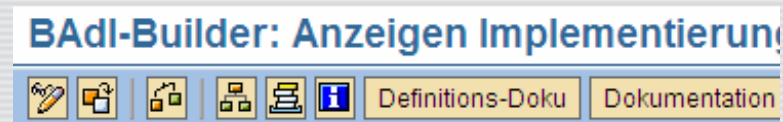
- Problem: Ermittlung Eintrittsdatum muß eigenen Infotypen beeinhalt
- Suche im SAPNET erfolglos
- Suche im Customizing-Leitfaden:





Beispiel für BAdI 2

Einbauhilfen über Definitions-Doku



HY SIMGHRPAD00_ENTRY_LEAVE

Kurztext

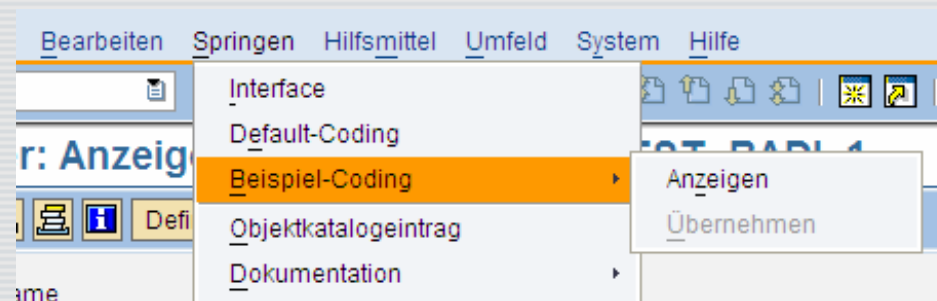
BAdI: Ermittlung der Eintritts-/Austrittsdatümer

Verwendung

Dieses Business Add-In (BAdI) wird in der Komponente Personaladministration (PA) verwendet.

Mit diesem Business Add-In können Sie nach Ihren individuellen Anforderungen Eintritts- und/oder Austrittsdaten eingetragener Mitarbeiter (Mitarbeiter) ermitteln.

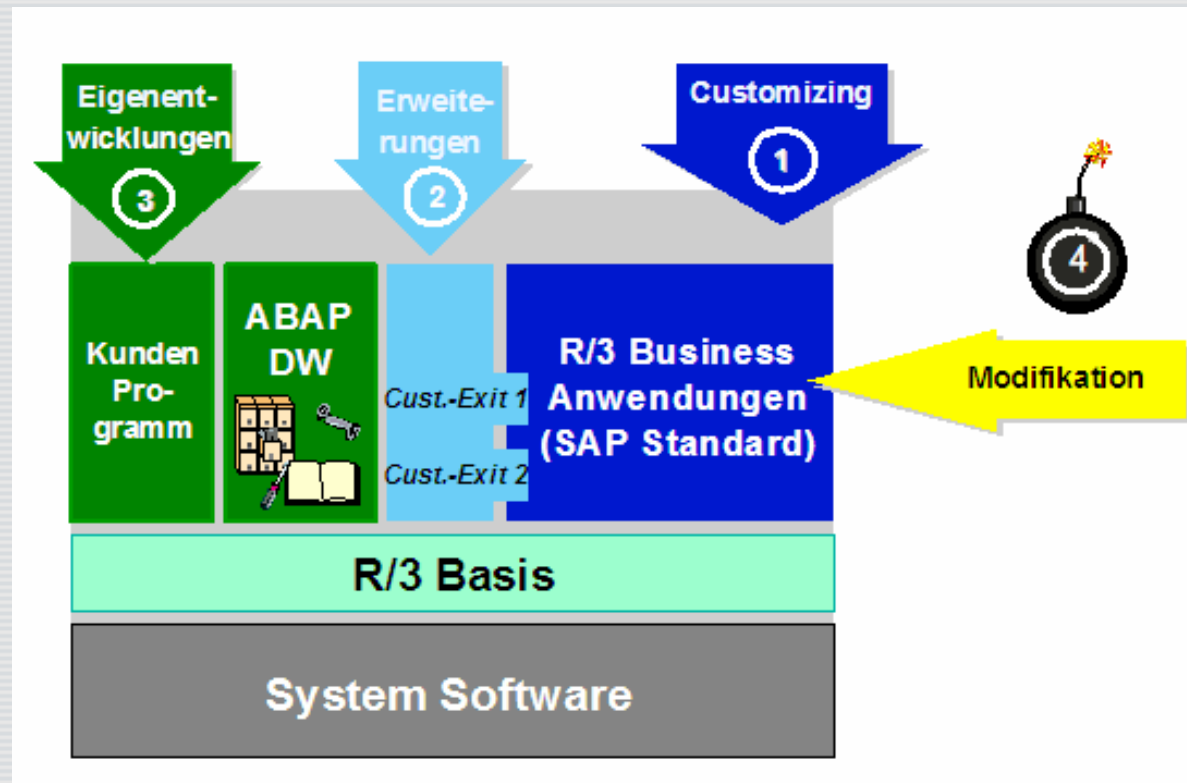
Beispiel-Coding über Menü





Beispiel: Modifikationassistent 1

- Problem: Der Entgeltnachweis darf beim Druck nicht gedreht werden.





Beispiel: Modifikationsassistent 1

- ≡ Vorteil: Standardprogramm kann weiterhin genutzt werden
- ≡ Nachteil: Modifikation muß immer wieder abgeglichen werden

Transaktionen SPAM / SPAU

```
*
*====> Editing payroll results for country germany    "<-- Change here
*
* 4.0C (AHR)
*XGFAHRK002587 30.01.98 Umstellung auf 9er Includes (not marked)
*
*#####
REPORT RPCEDTD0                                "<-- Change here
NO STANDARD PAGE HEADING
MESSAGE-ID 38
*{ REPLACE                                     1
*\ LINE-SIZE 132
  LINE-SIZE 80
*} REPLACE
  USING DATABASE PNP.

*-----
*  Allgemeiner Datenteil
*-----

INCLUDE RPCEDD09.
INCLUDE RPCEDBD0. "Datenteil nationale Besonderheiten
```



Beispiel: Eigene Funktion/ Operation 1

- Problem: Abrechnung muß eigenen Infotyp auswerten.
- Transaktion PE04 oder über Customizing Werkzeuge
- Kundennamensraum: & () - ; < >

Objekt Bearbeiten Springen Hilfsmittel System Hilfe

Pflege Funktionen und Operationen

Alle aktivieren Eingabeparameter Ausgabeparameter

Name

Objektklasse
☒ Abrechnung
☐ Zeitwirtschaft

Objekttyp
☒ Funktion
☐ Operation

Anlegen Ändern Anzeigen Löschen



Beispiel: Eigene Funktion/ Operation 2

- ≡ Vorteil: in Abrechnungsschemen /-regeln einbaubar
- ≡ Vorteil: eigener Bereich im Report, daher keine Modifikation

```
PROGRAMM  Bearbeiten  Springen  Hilfsmittel  Umfeld  System  Hilfe

ABAP Editor: Include ZP01IL_ABR_FUNKT_OPER

Include      ZP01IL_ABR_FUNKT_OPER      aktiv

FORM fuz9250.

  DATA: aus-begda TYPE d.
  DATA: aus-endda TYPE d.
  DATA: sel-begda TYPE d.
  DATA: sel-endda TYPE d.
  DATA: h-datum TYPE d.

  DATA: returncode LIKE sy-subrc.
  DATA: message_text LIKE error_ptext-text1.

  REFRESH: p9250, izzhr_lak_lga, z9250.
  CLEAR:   p9250, izzhr_lak_lga, zzz_eiart, zzz_wann.
  rp-read-infotype pernr-pernr 9250 p9250 pn-begda pn-endda.

  LOOP AT p9250 WHERE endda GE pn-begda
                        AND   begda LE pn-endda.
  ENDLOOP.
```



Unser Team steht Ihnen bei der
Umsetzung gerne hilfreich zur
Seite oder übernimmt die
Umstellung für Sie.

Vielen Dank für Ihre
Aufmerksamkeit !



Kontakt



Peter Klimke
Senior Consultant ERP

Stieghorster Straße 60
33605 Bielefeld

Fon 0521 2 08 89 10
Mobil 0172 5 21 72 06

www.gedoplan.de
Peter.Klimke@gedoplan.de

involva gruppe