



Enhancement Framework

Das neue Erweiterungskonzept

HR-User-Group 27.05.2009



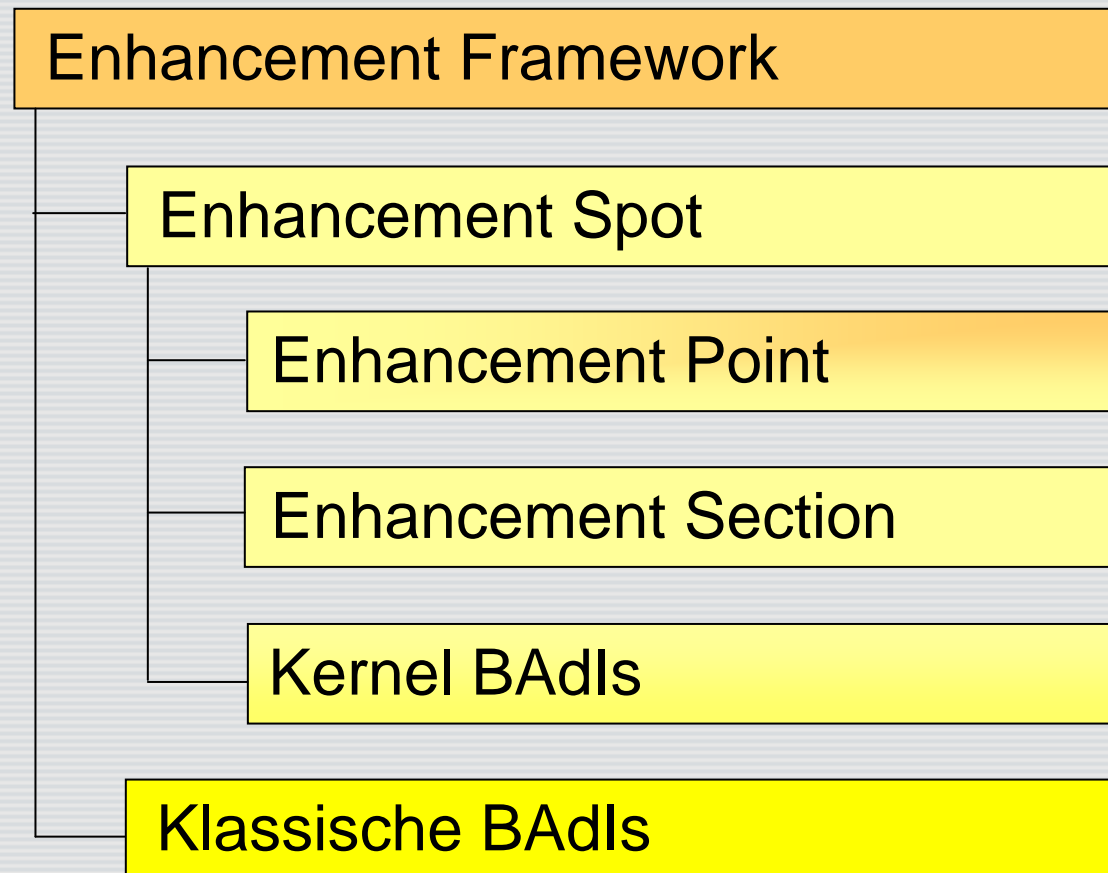
Anpassungsmöglichkeiten bisher

- ≡ Eigenentwicklung
- ≡ User-Exit
- ≡ Customer-Exit
- ≡ Business Transaction Events (BTE)
- ≡ Business AddIn (BAI)
- ≡ Modifikation

Problem : Unterschiedlichste Techniken
 Jeweils eigene Implementierungswerkzeuge



Das neue Erweiterungskonzept (Enhancement Framework)





Enhancement Points

- ≡ Ziel : Modifikationsfreies Einfügen von Quelltexten, Variablen- und Parameterdeklarationen in SAP-Programmen, SAP-Funktionsbausteinen und SAP-Methoden.
- ≡ Unterschieden werden:
 - ≡ *Expliziter* Enhancement Point
Von SAP vorbereitete Einfügemöglichkeit
 - ≡ *Impliziter* Enhancement Point
an bestimmten Stellen standardmäßig, also ohne Vorgabe durch SAP-Entwickler Erweiterungsmöglichkeit.



Expliziter Enhancement Point

- Einfügepunkt im Coding SAP-seitig vorgedacht.
- Zu Enhancement Point kann Implementierung angelegt werden.
- Eventuell bereits vorhandene Implementierungen können redefiniert (ersetzt) oder zurückgenommen werden.
- Mehrfache Implementierungen zum gleichen Enhancement Point sind möglich.

```
l_curr_forced = 'K'.      "Default ist Kostenrechnung
IF flg_waehrung = cnt_fwk.  "Wenn Fremdwährung
    l_curr_forced = 'O'.    "So wird Objektwährung
ENDIF.                    "END P40K046473

* Drucken über Einzelpostenbericht
ENHANCEMENT-POINT FCOD_PRIN_01 SPOTS ES_SAPMK70D STATIC.
*$$$-Start: FCOD_PRIN_01-----
ENHANCEMENT 1  MGV_MATNR_LAMA_SAPMK70D.    "active version
* MANU: KKB_MATNR Domain change
    data: rxm0_kkbmatnr like KEKO-MATNR.
ENDENHANCEMENT.
```

Nächste gleiche Strukturstufe	
Vorige gleiche Strukturstufe	
Gehe zu Start der Einheit	
Gehe zu Ende der Einheit	
Block/Ablage	
Erweiterungsimplementierung	Anlegen
	Zurücknehmen
	Ersetzen
	Ändern



Implementierung anlegen

* Drucken über Einzelpostenbericht

ENHANCEMENT-POINT FCOD_PRIN_01 **SPOTS** ES_SAPMK70D **STATIC.**

*\$\$\$-Start: FCOD_PRIN_01

Druck/Anlage

Erweiterungsimplementierung

Anlegen

Zurücknehmen

Erweiterungsimplementierung anlegen

Erweiterungsimplementierung

Z_MK70DF20_MS

Kurztext

Erweiterung Ausdruck Einzelposten

Zusammengesetzte Erweiterungsimplementierung



* Drucken über Einzelpostenbericht

ENHANCEMENT-POINT FCOD_PRIN_01 **SPOTS** ES_SAPMK70D **STATIC.**

*\$\$\$-Start: FCOD_PRIN_01

ENHANCEMENT 4 Z_MK70DF20_MS.

data :

ENDENHANCEMENT.

ABAP Editor: Erweiterung Z_MK70DF20_MS ändern

← → Aktiv <-> Inaktiv Erweiterungen Muster Pretty Print

Include

MK70DF20

Aktivieren Erweiterungen (Strg+F3)



Impliziter Enhancement Point

- Erweiterungsstelle nicht von SAP vorgedacht, aber dennoch nicht frei wählbar.
- Zulässige Stellen für implizite Source Code Plugins können im Editor angezeigt werden.

The screenshot shows the SAP Source Code Editor interface. On the left, the 'Bearbeiten' (Edit) menu is open, and the path 'Bearbeiten' > 'Erweiterungsoperationen' > 'Implizite Erw.-Optionen einblenden' is highlighted with orange boxes. Red arrows point from these menu items to the corresponding lines in the code snippet on the right.

```
4788  form check_texte using bukrs belnr gjahr xvorb.  
4789  ~~~~~  
4790  clear xvorb.  
4791  call function 'FI_TEXTS_DOCUMENT'  
4792  exporting  
4793      i_belnr      = belnr  
4794      i_bukrs      = bukrs  
4795      i_check      = 'X'  
4796      i_gjahr      = gjahr  
4797      i_object     = 'BELEG'  
4798  exceptions  
4799      no_texts_found = 1  
4800      others         = 2.  
4801  if sy-subrc = 0.  
4802      xvorb = 'X'.  
4803  endif.  
4804  ~~~~~  
4805  endform.                                "CHECK_TEXTE
```



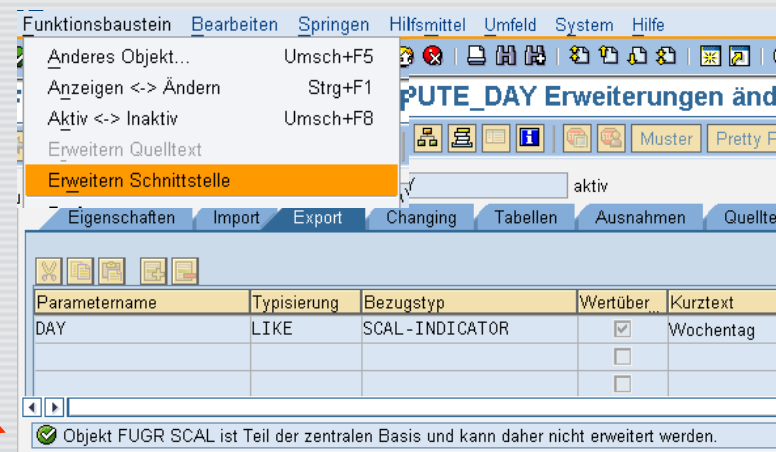
Implizite Enhancement Points

- Am Ende einer Strukturdefinition vor „END OF ...“ zur Aufnahme zusätzlicher Komponenten.
- Am Anfang und am Ende von Unterprogrammen, Funktionsbausteinen und Methoden lokaler bzw. globaler Klassen zum Einfügen zusätzlichen Codings.
- Am Ende des IMPORTING-/EXPORTING-/CHANGING-Deklarationsblocks von Methoden lokaler Klassen zur Aufnahme zusätzlicher Parameter
- Am Ende der Public/Private/Protected Section einer lokalen Klasse zur Definition zusätzlicher Attribute und Methoden
- Am Ende des Implementation-Block lokaler Klassen zur Implementierung der zusätzlich deklarierten Methoden



Implizite Enhancement Points

- In Schnittstellen-Definitionen von Funktionsbausteinen und Methoden globaler Klassen zur Aufnahme zusätzlicher Parameter Zu globalen Klassen können beliebige Attribute und Methoden hinzugefügt werden.
- Zu Methoden von globalen Klassen können so genannte Pre- und Post-Methoden definiert werden, die automatisch ausgeführt werden.
- SAP-Methoden können durch Overwrite-Methode ersetzt werden.
- Am Ende von Includes



Aber Achtung, es
gibt Ausnahmen !



Enhancement Section

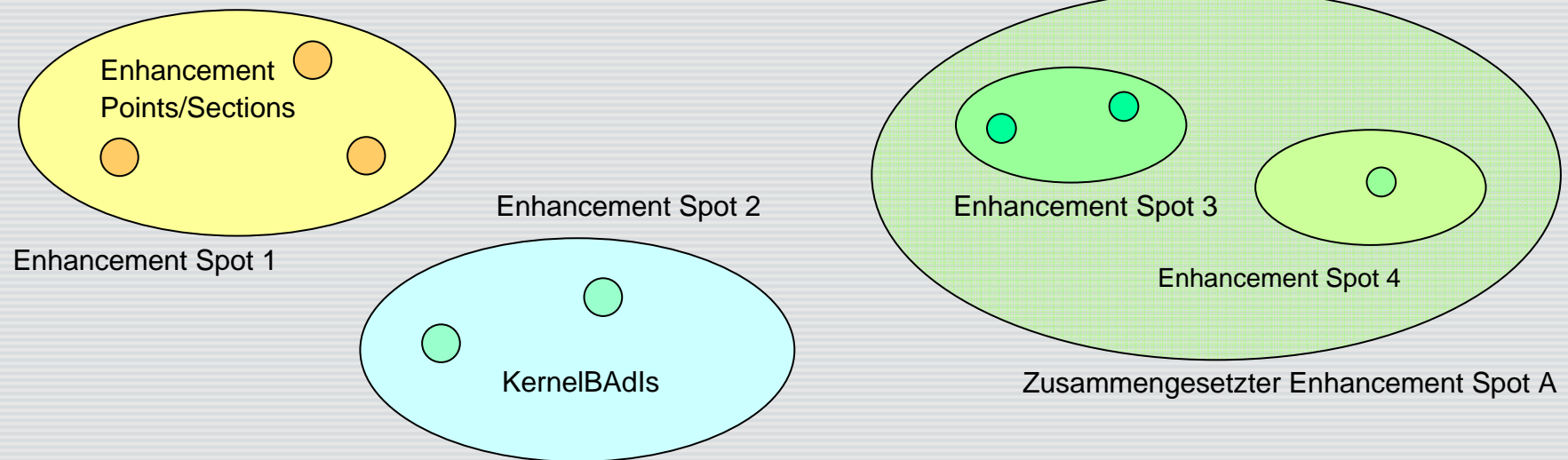
- Möglichkeit zur modifikationsfreien Ersetzung von SAP-Quelltext in Programm, Funktionsbausteinen und Methoden.
- Es gibt keine impliziten Enhancement Points

```
enhancement-section      fcod_prin_03 spots es_sapmk70d.  
  call function 'K_KKB_ITEMIZATION'  
    exporting  
      i_kdauf          = 1_kdauf  
      i_kstrg          = 1_kstrg  
      i_tvers          = 1_tvers  
      i_gjahr          = 1_gjahr  
      i_pronr          = 1_pronr  
      i_aufnr          = 1_aufnr  
    exceptions  
      input_incomplete      = 1  
      input_not_unique      = 2  
      no_calculation_found  = 3  
      no_items_found        = 4  
      wrong_input           = 5  
      others                 = 6.  
-end-enhancement-section.
```



Enhancement Spots

- Explizite Enhancement Points, Enhancement Sections und Kernel BAdIs werden mit Hilfe von Enhancement Spots zusammengefasst und verwaltet.
- Zusammengesetzte Enhancement Spots können elementare Enhancement Spots und andere zusammengesetzte Spots bündeln.





Auffinden von Enhancement Spots und Implementierung

- Enhancement Spots können mit Hilfe des „Erweiterungsinfosystems“ in der SE80 ermittelt und Implementierungen verwaltet werden

Erweiterungsspot /BEV1/ES_EMAUSW anzeigen

Erweiterungs-Infosystem
Repository Browser
Repository Infosystem

Erweiterungen
*

Objektname	Beschreibung
↳ Gesamtsicht Erweiterungen	
↳ zugses. Erw.Impl. Parents	
↳ RE	RE-FX (Flexibles Immobilienmanagement)
↳ EC_ISHMED	i.s.h.med Erweiterungsimplementierungen
↳ EC_ISH	IS-H Erweiterungsimplementierungen
↳ Spot-Composites Parents	
↳ ESC_ISHMED	
↳ ESC_ISH	
↳ DFPS_COMPOSITE	
↳ zugses. Erw.Spots. ohne Paren	
↳ Erw.Spots. ohne Paren	
○ /BEV1/ES_EMAUSW	
○ /BEV1/ES_EMLSALD	
○ /BEV1/ES_EMLSALD_SF	



Neue BAdI-Technik (Kernel-BAdIs)

- ≡ Ziel
 - ≡ Integration in das Enhancement Framework
 - ≡ Integration in das Switch Framework
 - ≡ Performance-Steigerung
 - ≡ Realisierung von Zusatzfunktionen
- ≡ Klassische BAdI-Technik noch vorhanden, wird aber nicht mehr weiterentwickelt
- ≡ Migration klassischer BAdIs möglich



Klassische BAdI-Technik

- Instanz der Adapterklasse erzeugen
- Interface-Methode aufrufen, die gleichnamige Methoden aktiver Implementierungen aufruft.

```
DATA : wa          TYPE spfli,  
      r_badi       TYPE REF TO zif_ex_badidemo.  
  
CALL METHOD cl_exithandler=>get_instance  
  CHANGING  
    instance = r_badi.  
  
□ SELECT * FROM spfli INTO wa.  
  WRITE : / wa-carrid, wa-connid.  
  CALL METHOD r_badi->display_more  
    EXPORTING  
      spfli_data = wa.  
ENDSELECT.
```



Neue BAdI-Technik

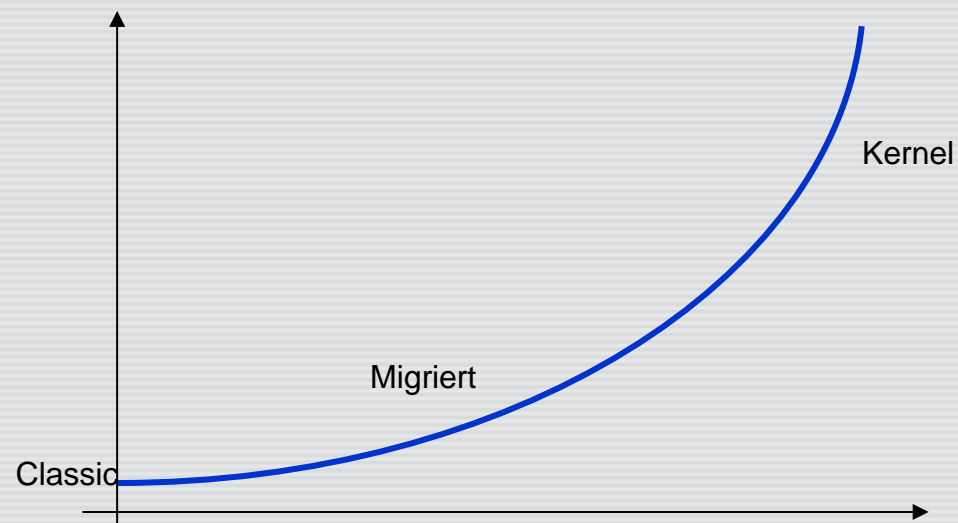
- ≡ BAdI-Handle beschaffen (im Kernel vorhanden)
- ≡ Methode des BAdI-Handles aufrufen, die gleichnamige Methoden aktiver Implementierungen aufruft.

```
DATA : wa      TYPE spfli,  
      r_badi   TYPE REF TO ZBADIDEMO_ES_BADI.  
  
get badi r_badi.  
  
□ SELECT * FROM spfli INTO wa.  
  WRITE : / wa-carrid, wa-connid.  
  CALL BADI r_badi->display_more  
    EXPORTING  
      spfli_data = wa.  
ENDSELECT.
```



Performancevergleich

- ≡ Classic BAdI Faktor 1
- ≡ Migriertes BAdI Faktor 2 – 27 schneller
- ≡ Kernel BAdI Faktor 40 – 600 schneller





Switch Framework (Zuordnung von Objekten zu Schalter)

Schalter Definition

Schalter	Systemweiter Status	Kurzbeschreibung
ZDEMO	off	Demo
ZJWO	on	
ZMSP	on	

Pakete | GUI / Dynpro | Bereichsmenü Knoten

Schalter	Paket	Kurzbeschreibung	Aktiv / Inaktiv
ZDEMO	ZDEMOSW	Paket für Switch	Aktiv

Dynpronummer: 100 | aktiv(überarbeitet)

Eigenschaften | Elementliste | Ablauflogik

H...	MName	Typ d...	Schalter	Reaktion auf Schalter
	SCUSTOM-ZZFAX	Text	ZDEMO	Anzeigen
	SCUSTOM-ZZFAX	I/O		Ausblenden
		OK		

Funktionseigenschaften

Funktionscode: HALLO

Funktionstyp: ☐ Anwendungsfunktion

Schalter: ZDEMO

Reaktion: ☐ Anzeigen ☐ Ausblenden

ZDEMOSW

Eigenschaften | Verwendungserklärungen | Paketschlüssel

Kurzbeschreibung	Paket für Switch
Verantwortlicher	ADMIN
Anwendungskomponente	
Schalter	ZDEMO

ZDEMOSW

Eigenschaften | Erw.-Implementierungen | Technisch

Erweiterungstechnik	Object-Plugin (BAdI)
Beschreibung	Demo
Paket	ZDEMOSW
Schalter	ZDEMO
Schalterstellung	F

ZDEMOSW

Append-Struktur | ZASCUST

Kurzbeschreibung	Zusatzfelder
------------------	--------------

Eigenschaften | Komponenten | Eingabehilfe/-prüfung

Letzte Änderung	ADMIN	08.03.2007
Paket	ZDEMOSW	
Appendierende Tab.	SCUSTOM	
Schalter	ZDEMO	

Einem Schalter werden Pakete, Dynproelemente, Menü-Einträge, ... zugeordnet



Switch Framework (Business Functions und Business Function Sets)

Business Function Definition

Business Function	Kurztext
ZDEMOFUNC	Demofunktion
Z.IWO	.IWO-Test

Switches Ausgeschlossene Business Function Sets

Business Function	Schalter	Typ	Kurztext
ZDEMOFUNC	ZDEMO	Aktivierung	Demo-Schalter
ZDEMOFUNC	ZMSP	Aktivierung	Demo-Switch

Einer Business-Function werden Schalter zugeordnet

Business Function Set

Business Function Set	Kurztext	Anleger
MINING	SAP Mining	SAP
ZDEMOBFS	Demo Business Function Set	ADMIN

Business Functions Redefinitionskonflikte BADI Impl. Konflikte

Business Function Set	Business Function	Nicht anzeigen
ZDEMOBFS	ZDEMOFUNC	<input type="checkbox"/>
ZDEMOBFS		<input type="checkbox"/>

Ein Business-Function-Set fasst Business-Functions zusammen und wird bei Bedarf aktiviert.



Weitere Infos

ABAP-Schlüsselwortdokumentation

ABAP - Schlüsselwortdokumentation

- ABAP - Übersichtsartikel
- ABAP - Referenz
 - ABAP-Syntax
 - Erweiterungen
 - Quelltext-Erweiterungen
 - Implizite Erweiterungsoptionen**
 - Explizite Erweiterungsoptionen
 - Erweiterungsimplementierung
 - ENHANCEMENT
 - ENDENHANCEMENT
 - Erweiterungen über BADs
- ABAP System-Klassen und -Interfaces
- ABAP Konzepte

Implizite Erweiterungsoptionen

Neben explizit mit [ENHANCEMENT-POINT](#) und [ENHANCEMENT-SEC](#) implizite Erweiterungsoptionen vorhanden:

- hinter der letzten Zeile des Quelltexts von ausführbarer
- vor der ersten und hinter der letzten Zeile der Im
- am Ende eines Sic
- am Ende eines

SAP **SAP NETWORK**
Welcome Manfred Sprenger

SDN Community | BPX Community | Forums | Wiki | Blogs | Downloads

Search

Enhancement

All

Advanced Search

Search In

Search Results For Enhancement (~9533 results)

[Enhancement Framework - The New Way to Enhance](#)
Presentation 460.3 kB 5/1/06 Oliver Mayer

Enhancement Framework -... HTML Version | [Properties](#) | [See Also](#)

Note 964900



Vielen Dank für Ihre Aufmerksamkeit!