



# **Enhancement Framework**

**- The new way to enhance your ABAP systems**

**Oliver J. Mayer**

**Product Manager – SAP NetWeaver**

### **As a result of this session, you will be able to:**

- **Understand the fundamental idea of the Enhancement Framework and Switch Framework (available in **SAP NetWeaver 2004s**)**
- **Reduce TCO by using enhancement technologies instead of modifications**
- **Enhance SAP standard objects**
- **Understand how Enhancement definitions are created**



**Motivation & Overview**

**Enhancement Framework**

**Source Code Plugin - Technology**

**Function Group Enhancement - Technology**

**Class Enhancement - Technology**

**Kernel-BAdl – Technology**

**Switch Framework**

**Summary**



## Motivation & Overview

### Enhancement Framework

Source Code Plugin - Technology

Function Group Enhancement - Technology

Class Enhancement - Technology

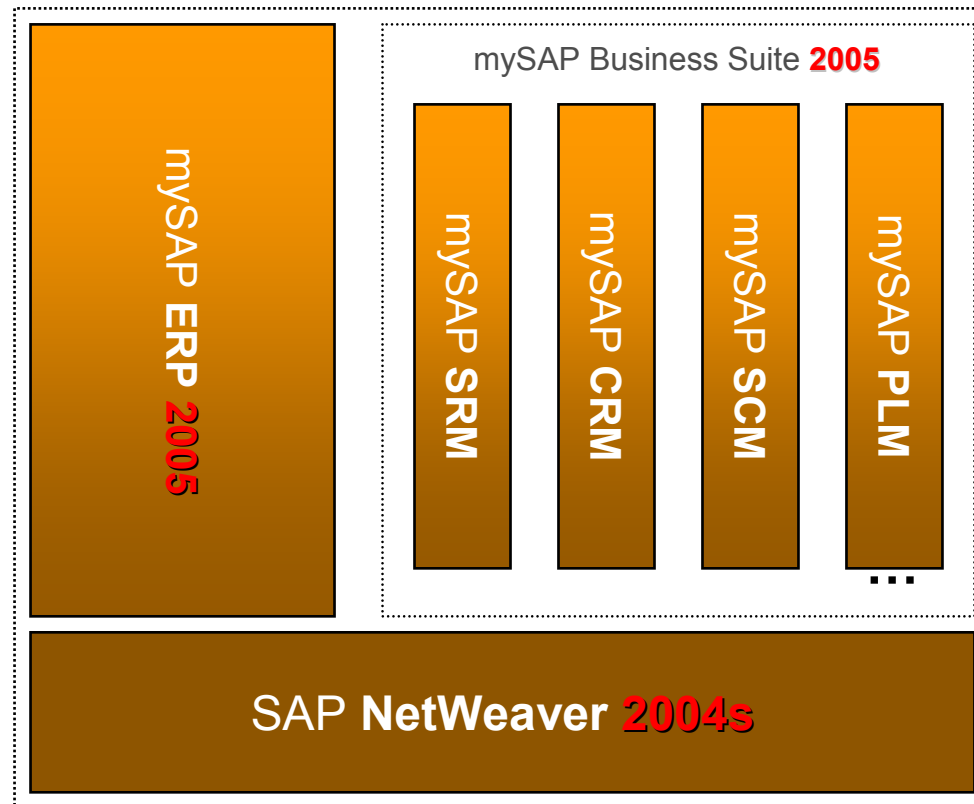
### Kernel-BAdl – Technology

### Switch Framework

### Summary

## App Server Release

... 4.6c 6.10 6.20 NW04 **NW04s**



# SAP NetWeaver

User Productivity Enablement	Running an Enterprise Portal	Enabling User Collaboration	Business Task Management	Mobilizing Business Processes	Enterprise Knowledge Management	Enterprise Search
Data Unification	Master-Data Harmonization	Master-Data Consolidation	Central Master-Data Management		Enterprise Data Warehousing	
Business Information Management	Enterprise Reporting, Query, and Analysis	Business Planning and Analytical Services	Enterprise Data Warehousing	Enterprise Knowledge Management	Enterprise Search	
Business Event Management	Business Activity Monitoring			Business Task Management		
End-to-End Process Integration	Enabling Application-to-Application Processes	Enabling Business-to-Business Processes	Business Process Management	Enabling Platform Interoperability	Business Task Management	
Custom Development	Developing, Configuring, and Adapting Applications			Enabling Platform Interoperability		
Unified Life-Cycle Management	Software Life-Cycle Management			SAP NetWeaver Operations		
Application Governance and Security Management	Authentication and Single Sign-On			Integrated User and Access Management		
Consolidation	Enabling Platform Interoperability	SAP NetWeaver Operations	Master-Data Consolidation	Enterprise Knowledge Management	Enterprise Data Warehousing	
ESA Design and Deployment	Enabling Enterprise Services					

**One of the advantages of SAP software is the possibility to adapt the software to own requirements and the possibility of keeping the adaptations during upgrade.**

### **Ways of adaptation:**

- Customizing
- Enhancement
- Modification

### Reducing TCO

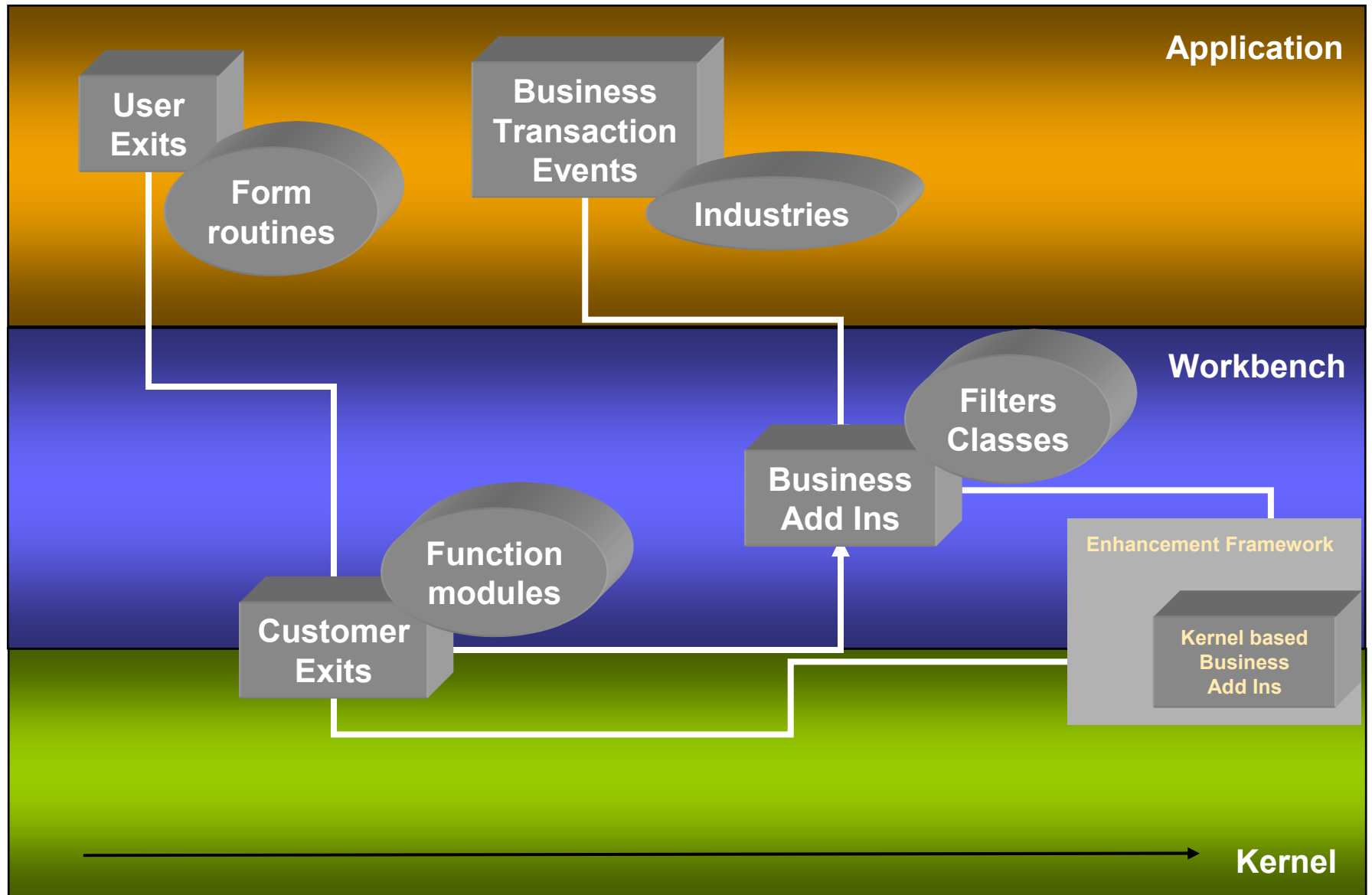
- **Enhancing objects instead of modifying them** reduces the effort for adjustment during SP import or upgrade.

### Disadvantages of modifications

- No support for **multiple users or projects**
- No support for **parallel developments**
- Will appear much more often in adjustment tools
- **Higher** adjustment **effort** (during upgrade & SP import)



# Evolution of SAP Enhancement Technology





Motivation & Overview

**Enhancement Framework**

Source Code Plugin - Technology

Function Group Enhancement - Technology

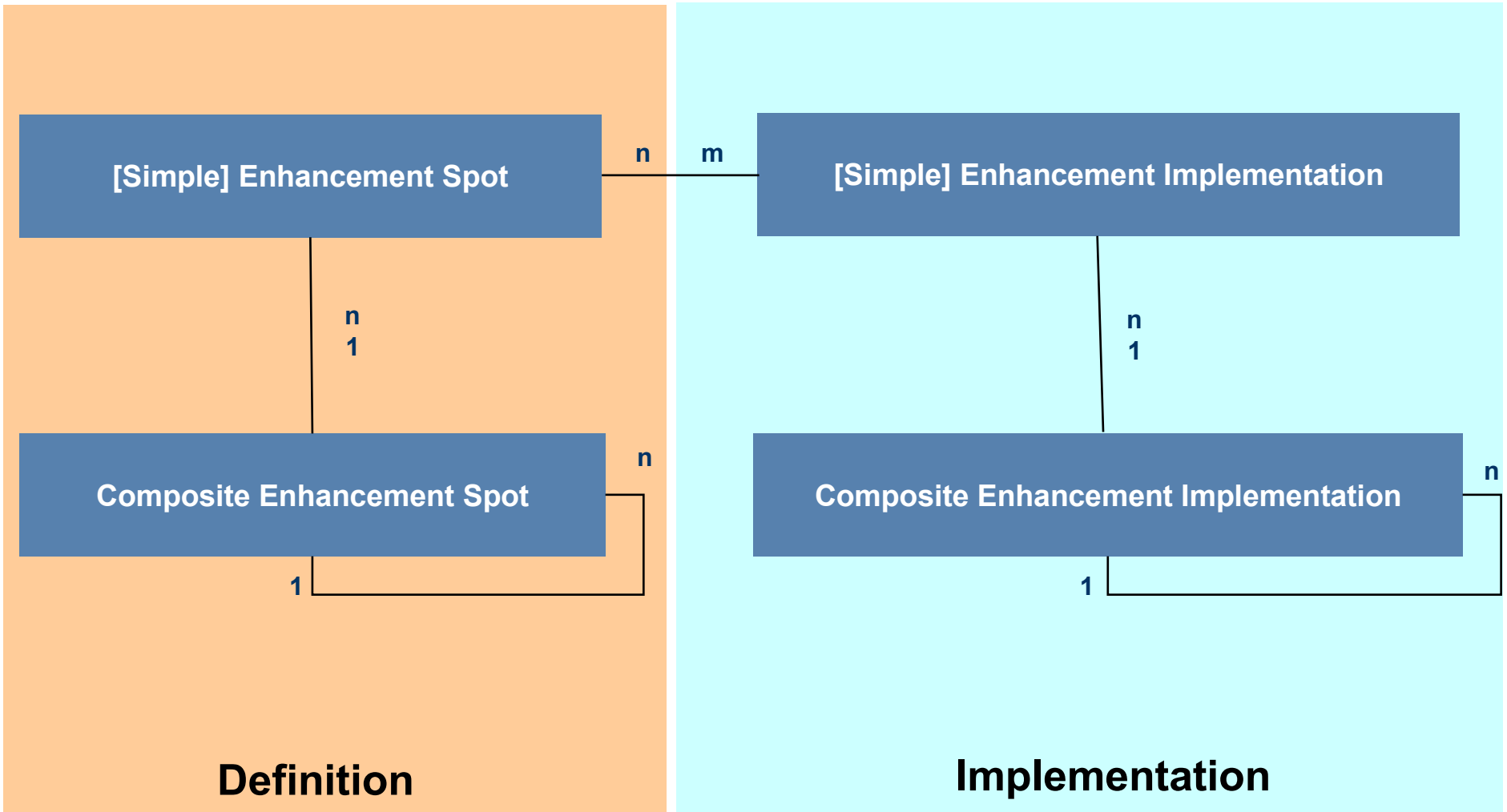
Class Enhancement - Technology

Kernel-BAdl – Technology

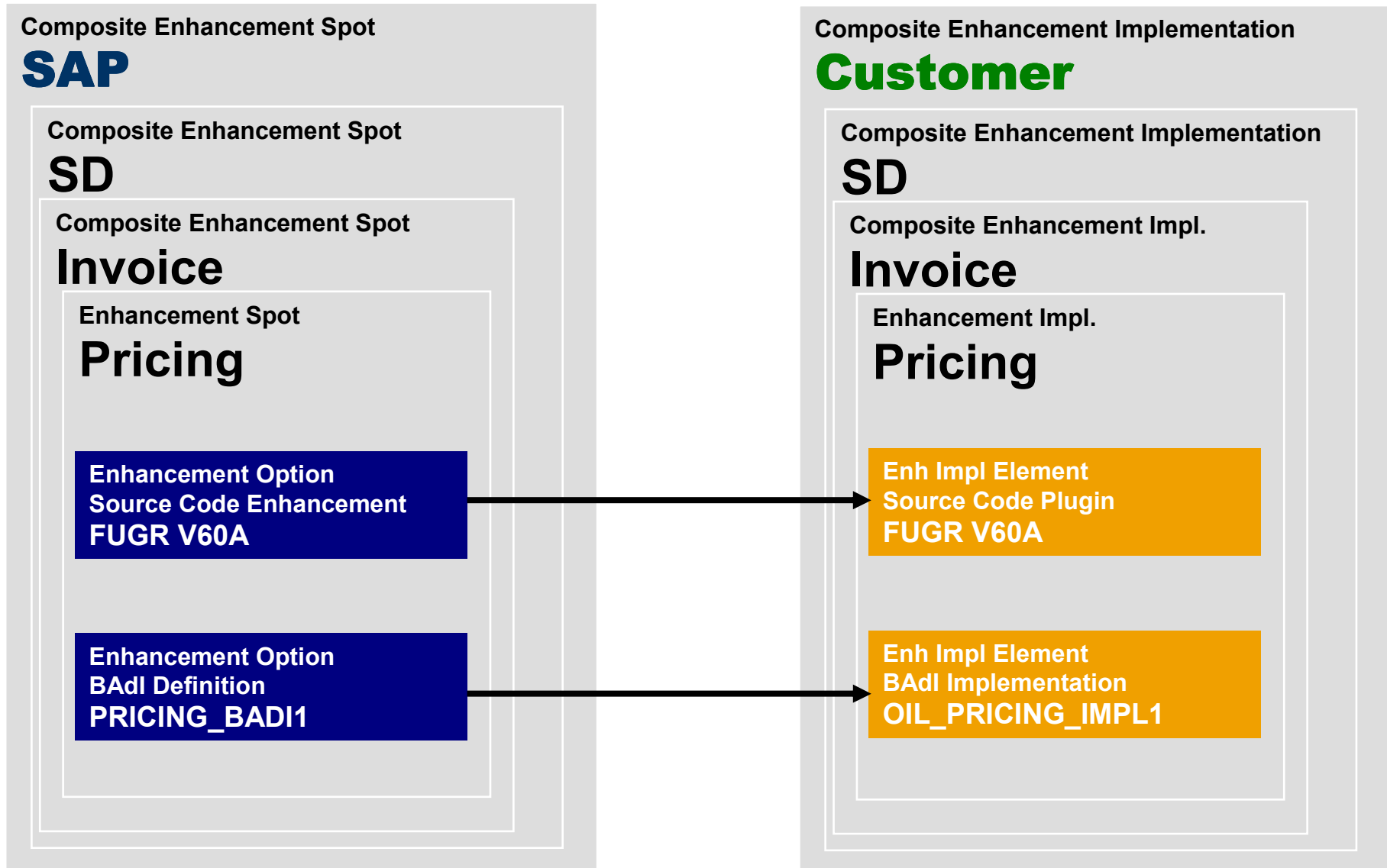
Switch Framework

Summary

# Enhancements - Relations



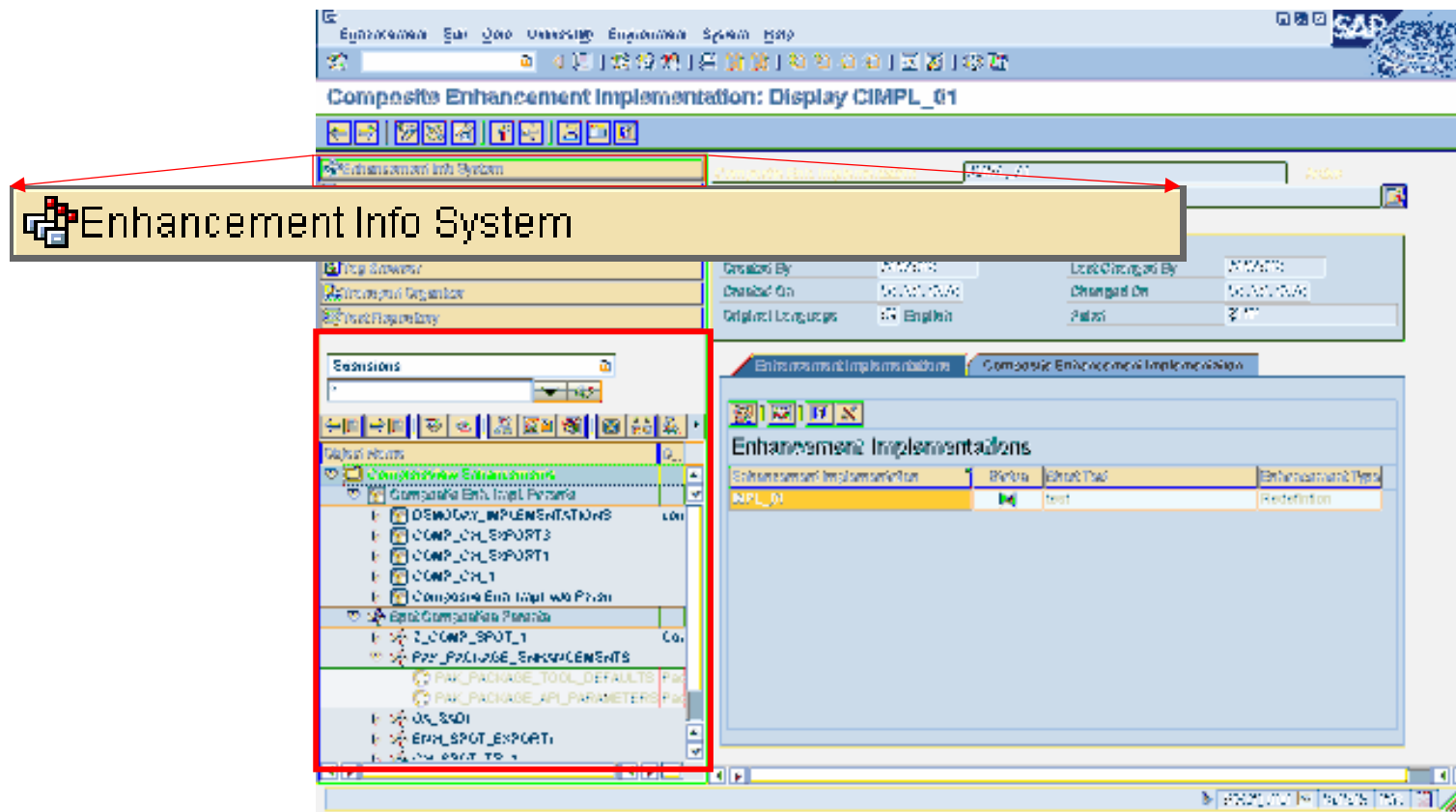
# Terminology by Example



# Enhancement Browser

## Search for

- Enhancements possibilities (Definitions – typically provided by SAP)
- Enhancement Implementations (typically done by Customer)



Integrated into Object Navigator (SE80)



Motivation & Overview

**Enhancement Framework**

**Source Code Plugin - Technology**

Function Group Enhancement - Technology

Class Enhancement - Technology

Kernel-BAdl – Technology

Switch Framework

Summary

# Modification-free enhancement of source code

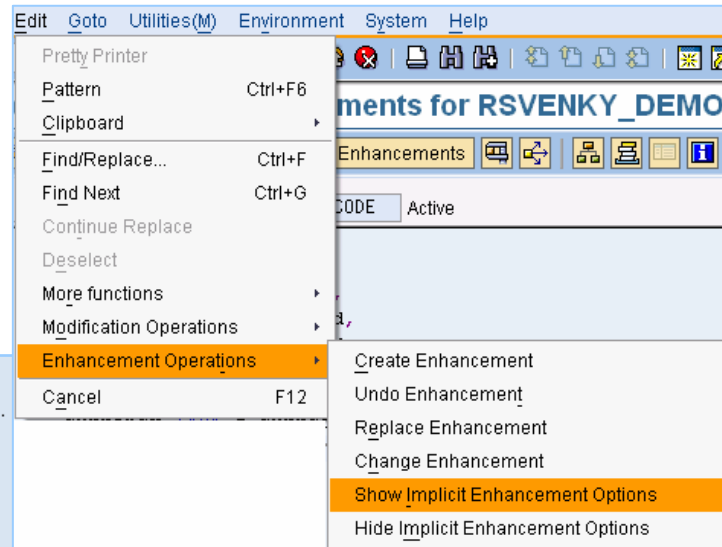
## Implicit Enhancement Option

- At common enhancement places, implicit Enhancement options are available. Examples:
  - ◆ End of Executable Program, Include, Function group, Dialog module
  - ◆ Begin/End of Form routine / Function module / Method
  - ◆ End of a structure
  - ◆ End of Private/Protected/Public Section of a local class
  - ◆ ...

## Explicit Enhancement Option

- Predefined enhancement options can be defined in source code. They are additionally stored inside Enhancement Spots.

# Implicit Enhancement Options



```
* Listoutput
LOOP AT gt_flights INTO gs_flights.
WRITE: / gs_flights-carrid,
gs_flights-connid,
gs_flights-fldate,
gs_flights-price,
gs_flights-currency.
ENDLOOP.
```

```
*****$$\SE:(1 ) Include ZSEFW_00_FLIGHTS_DISPLA
```

```
METHOD GET_PROPERTY.
```

```
*****$$\SE:(4 ) Klasse CL_GUI_OBJECT, Methode G
```

```
* (PROPERTY, P1..P16, P_COUNT, QUEUE_ONLY; => VALUE)
```

```
* check handle
```

```
DATA : STRPROPNAME TYPE STRING.
```

```
DATA : STRVALUE TYPE STRING.
```

```
DATA : QUEUE_EMPTY TYPE CHAR01.
```

```
""
```

```
*****$$\SE:(3 ) Klasse CL_GUI_CONTROL, PUBLIC S
```

```
*****$$\SE:(2 ) Klasse CL_GUI_CONTROL, PROTECTE
```

```
*****$$\SE:(1 ) Klasse CL_GUI_CONTROL, PRIVATE
```



## Explicit Enhancement Options

```
* Selection screen
SELECT-OPTIONS:
  so_carr FOR gv_carrid,
  so_conn FOR gv_connid.
```

```
ENHANCEMENT-POINT FLIGHTS_DECLARATION SPOTS FLIGHTS_DISPLAY STATIC.
```

```
START-OF-SELECTION.
```

```
* Select Data
```

```
ENHANCEMENT-SECTION      FLIGHTS_DBSELECT SPOTS FLIGHTS_DISPLAY.
  SELECT carrid connid fldate price currency
    from sflight
    into table gt_flights
    where carrid in so_carr
      and connid in so_conn.
END-ENHANCEMENT-SECTION.
```

```
* Listoutput
```

```
  LOOP AT gt_flights INTO gs_flights.
    WRITE: / gs_flights-carrid,
           gs_flights-connid,
           gs_flights-fldate.
  ENDLOOP.
```

## Source Code Plugin Technology - Example


```
PROGRAM p1.  
  
WRITE 'Hello World'.  
  
ENHANCEMENT-POINT ep1 SPOTS  
s1.  
  
..  
..  
..  
  
ENHANCEMENT-SECTION ep2  
SPOTS s1.  
    WRITE 'Original'.  
END-ENHANCEMENT-SECTION.
```

```
ENHANCEMENT 1.  
    WRITE 'Hello  
Paris'.  
ENDENHANCEMENT.
```

```
ENHANCEMENT 2.  
    WRITE 'Hello  
London'.  
ENDENHANCEMENT.
```

```
ENHANCEMENT 3.  
    WRITE 'Enhanced'.  
ENDENHANCEMENT.
```

Use **Change Mode** for creating enhancement points & sections.

- use button  „Display <-> Change“ to switch to change mode.

Use **Enhancement Mode** for creating enhancement implementations.

- use button „**Change Enhancements**“  to switch to Enhancement mode
- use button „Display <-> Change“  to leave Enhancement mode

# **Demo**

## **Source Code Plugin**



Motivation & Overview

**Enhancement Framework**

Source Code Plugin - Technology

**Function Group Enhancement - Technology**

Class Enhancement - Technology

Kernel-BAdl – Technology

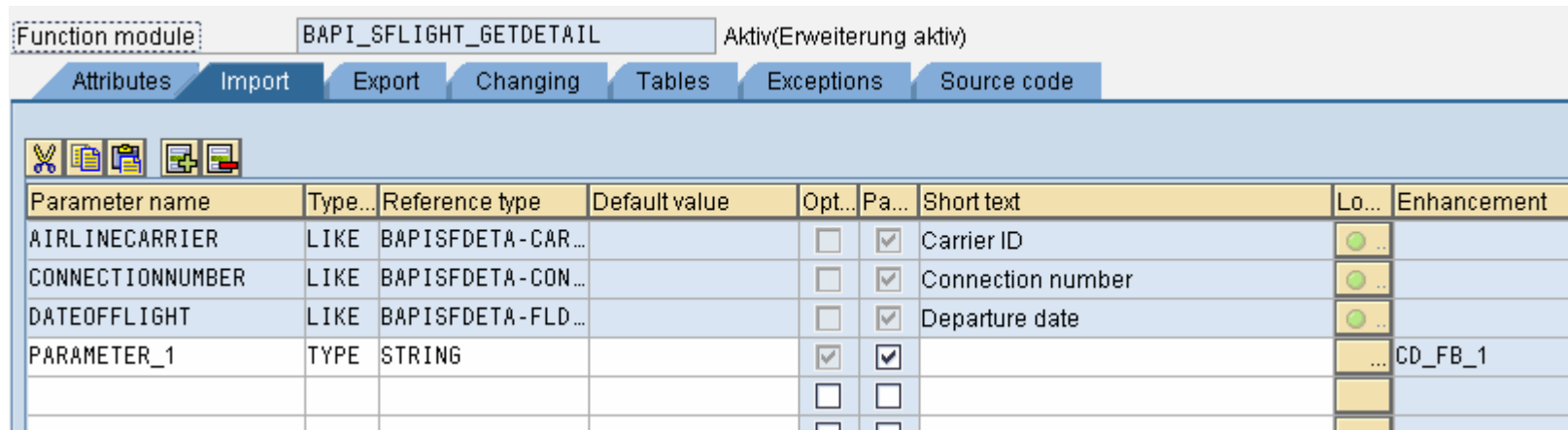
Switch Framework

Summary

# Function Group Enhancements

Function Group Enhancements allow:

- Adding new optional parameters to existing function modules



The screenshot shows the SAP SE37 transaction interface for the function module BAPI\_SFLIGHT\_GETDETAIL. The 'Import' tab is active, displaying a table of parameters. The table includes columns for Parameter name, Type, Reference type, Default value, Opt., Pa., Short text, Lo..., and Enhancement. The parameters listed are AIRLINECARRIER, CONNECTIONNUMBER, DATEOFFLIGHT, and PARAMETER\_1. The 'Enhancement' column shows 'CD\_FB\_1' for PARAMETER\_1.

Parameter name	Type...	Reference type	Default value	Opt...	Pa...	Short text	Lo...	Enhancement
AIRLINECARRIER	LIKE	BAPISFDETA-CAR...		<input type="checkbox"/>	<input checked="" type="checkbox"/>	Carrier ID	...	
CONNECTIONNUMBER	LIKE	BAPISFDETA-CON...		<input type="checkbox"/>	<input checked="" type="checkbox"/>	Connection number	...	
DATEOFFLIGHT	LIKE	BAPISFDETA-FLD...		<input type="checkbox"/>	<input checked="" type="checkbox"/>	Departure date	...	
PARAMETER_1	TYPE	STRING		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		...	CD_FB_1
				<input type="checkbox"/>	<input type="checkbox"/>			
				<input type="checkbox"/>	<input type="checkbox"/>			

Transaction: SE37

# **Demo**

## **Function Group Enhancement**



Motivation & Overview

**Enhancement Framework**

Source Code Plugin - Technology

Function Group Enhancement - Technology

**Class Enhancement - Technology**

Kernel-BAdl – Technology

Switch Framework

Summary



### **Class/Interface Enhancements allow addition of:**

- **optional parameters to existing methods**
- **methods**
- **events and event handlers**
- **references to interfaces**
- **Exits to existing methods**
  - ◆ **Pre-Exit – Called at the beginning of a method**
  - ◆ **Post-Exit – Called at the End of a method**
  - ◆ **Overwrite-Exit – Replaces the original method**

# Adding Methods & Parameters

## Adding new methods

Method	Level	Visibility	M...	Description	PreExit	PostExit	Enhancement
AFTER_IMPORT	Static	Public					
UPDATE	Static	Public					
READ	Static	Public					
WRITE	Static	Public					
ADD_METHOD_ONE	Instan	Private		additional functionality public			MATECHED2005_001
ADD_METHOD_PRIVATE	Instan	Private		additional functionality private			MATECHED2005_001

## Adding optional parameters to existing methods

Parameter	Type	P...	O...	Typing M...	Associated Type	Default value	Description	Enhancement
OBJ_NAME:	Importin	<input type="checkbox"/>	<input type="checkbox"/>	Type	TROBJ_NAME		Object Name in Object Directory	
PROTOCOL	Changir	<input type="checkbox"/>	<input type="checkbox"/>	Type	SPROT_U_TAB		Table Type for SPROT_U (Log In	
MY_ADD_PARAM	Importin	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Type	CHAR10		Characterfeld der Länge 10	MATECHED2005_001
MY_ADD_PARAM_EXP	Exportin	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Type	INT4		Natürliche Zahl	MATECHED2005_001

## Pre/Post Exits

...  
Call method instance->hugo( ).  
...

**Method Hugo.**

Coding.  
Coding.  
Coding.

**Endmethod.**

**Method Pre.**

...  
....  
**Endmethod.**

**Method Post.**

...  
....  
**Endmethod.**

# **Demo**

## **Class Enhancement**



Motivation & Overview

Enhancement Framework

Source Code Plugin - Technology

Function Group Enhancement - Technology

Class Enhancement - Technology

**Kernel-BAdl – Technology**

Switch Framework

Summary

### What are BADIs?

- **Business Add-Ins**
- is an anticipated **point of extension** – these points act like sockets and exist in the original coding
- has a **well-defined interface** in contrast to source code plug-ins and is therefore more stable to changes in the original coding

### Kernel BADIs - New Features

- Are **integrated** directly in the **ABAP Language/Runtime**
- **Improved filter** support allows non-character filter types (packed, numeric, string) and complex filter conditions
- Enable **reusable implementation** instances (Stateful BAdI)
- Control of the lifetime of implementations (BAdI-context)
- Allow for **inheritance** of implementations
- Can be **switched** by the Switch Framework

## Comparison: Usage of Old BAdIs vs. new BAdIs

### With Classic BAdI

```
DATA: bd TYPE REF TO if_intf.  
DATA: flt TYPE flt.
```

```
CALL METHOD c1_exithandler=>  
get_instance  
EXPORTING  
    exit_name = `BADI_NAME`  
CHANGING  
    instance = bd.
```

```
flt-lang = `D`.  
CALL METHOD bd->method  
EXPORTING  
    x          = 10  
    flt_val   = flt.
```

selecting implementations and issuing calls is mixed up

calls cause DB access

calls are redirected over a proxy class

### With New BAdI

```
data bd type ref to badi_name.  
get badi bd filters lang = `D`.  
call badi bd->method  
    exporting x          = 10.
```

selection occurs when the handle is requested

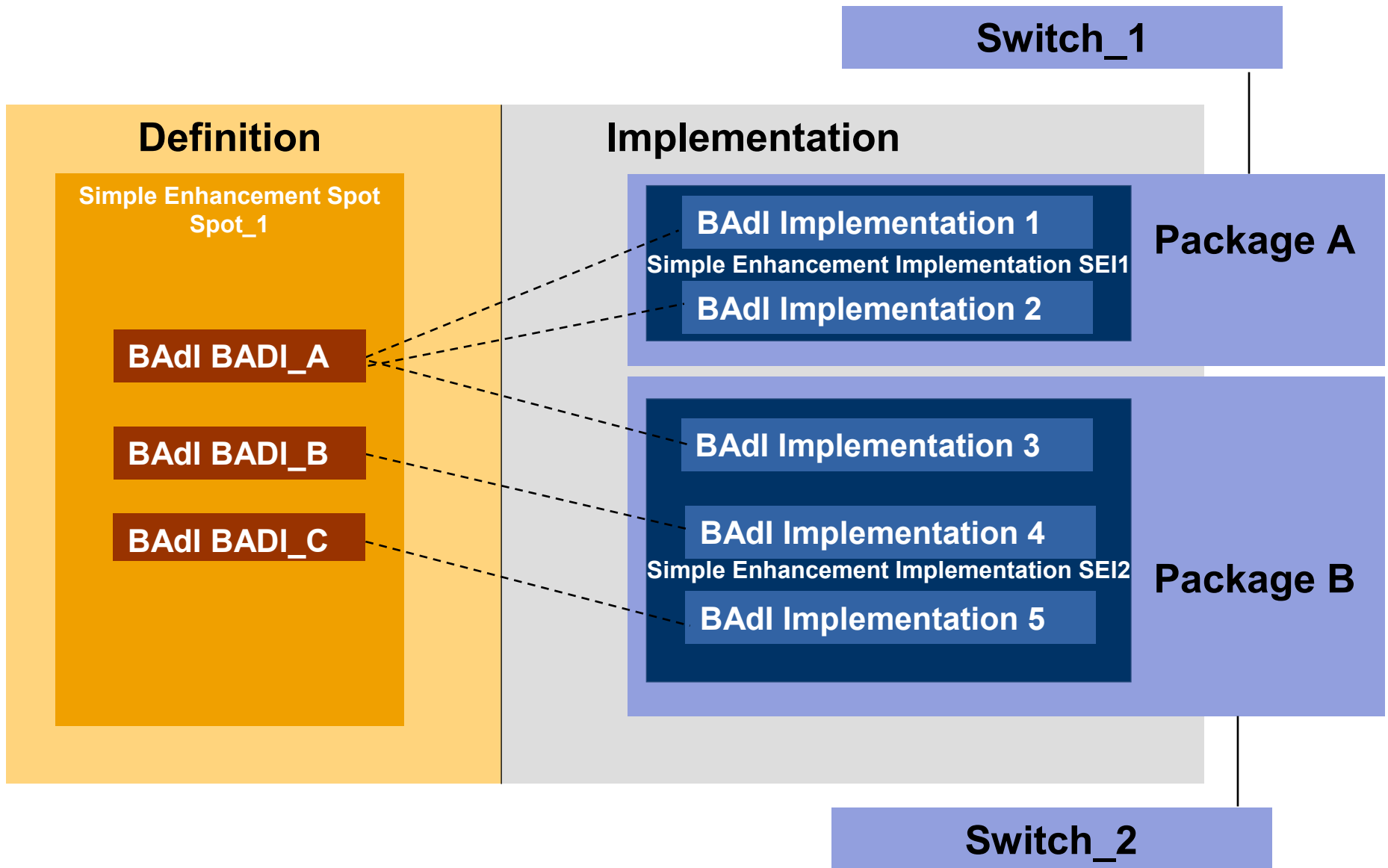
no DB access during runtime

Implementations are called directly (without a proxy)

Active implementations are evaluated at compile time and included in the load of the BAdI-handle.

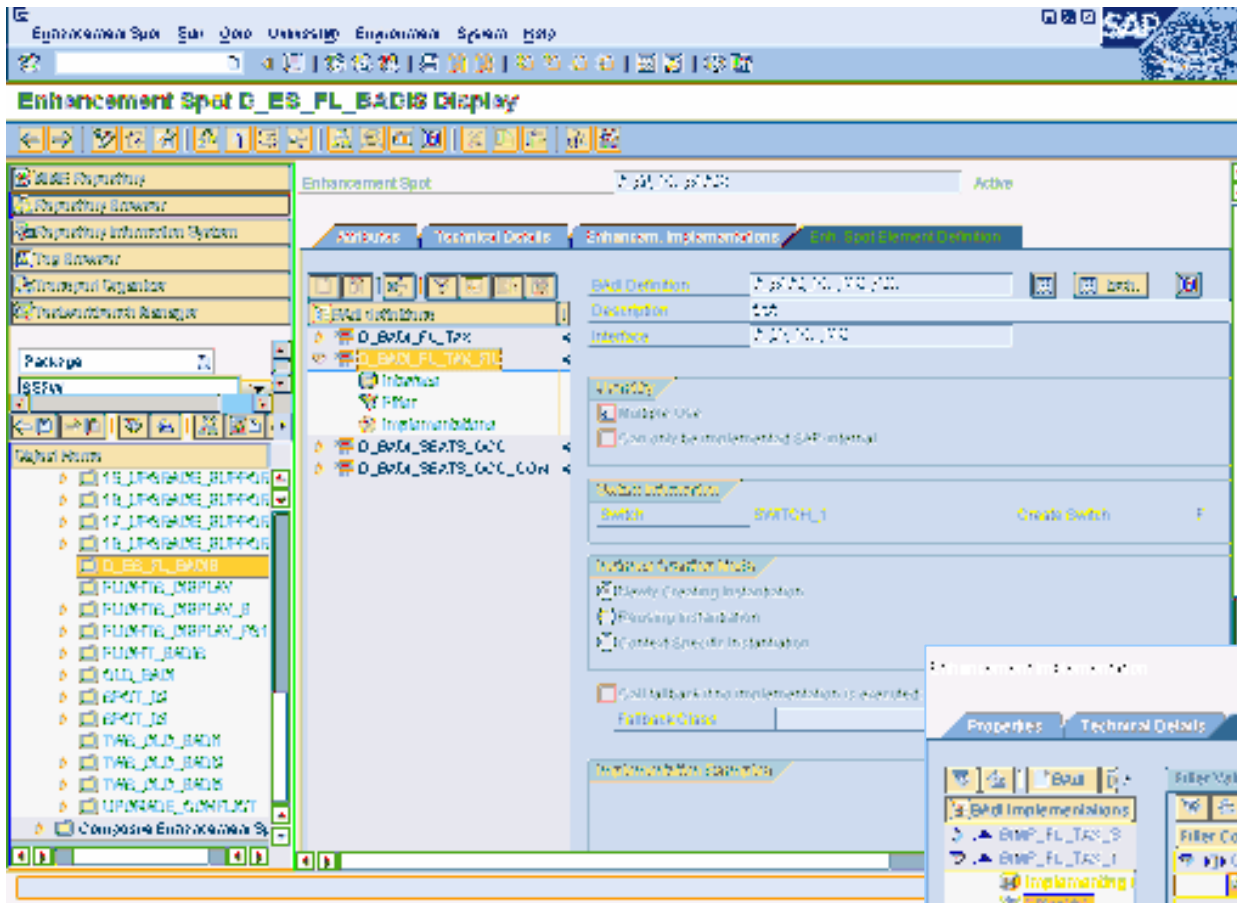
Old BAdIs are more expensive than the new ones.

# New BADI's and Enhancement Framework

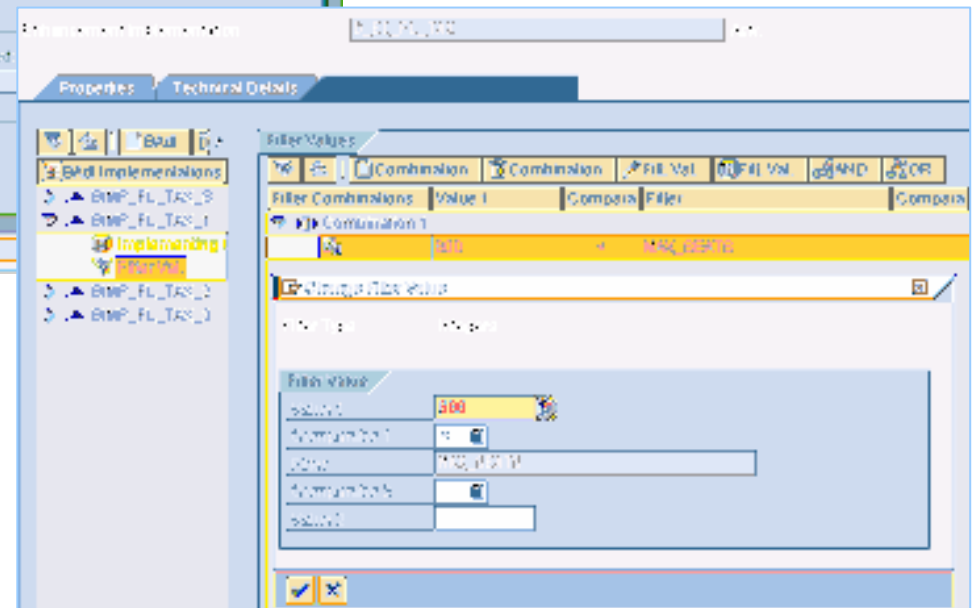




# Creating BADI in SE80



BADI Implementation (Creating Filters)



BADI Definition under Enhancement Spot

# **Demo**

## **Kernel-BAdI**

## BAdI Migration (Automatic Migration)

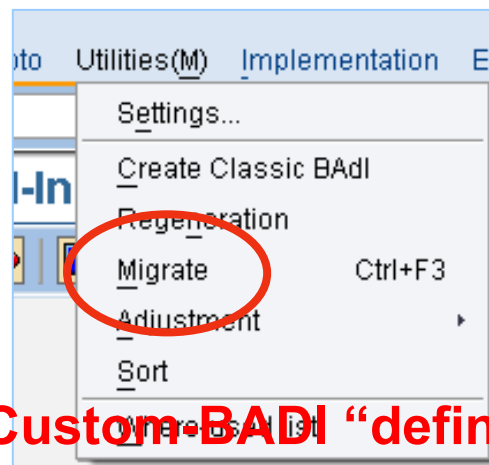
Automatic migration by selecting utilities → migrate from BAdI Builder (SE18)

- Specify Enhancement Spot for BAdI Definition
- Specify Enhancement Implementation for BAdI Implementation

→ no special knowledge necessary

→ effort: approximately 5 minutes per BAdI.

SE18 -> utilities -> Migrate



**ONLY when you have created Custom BADI “definitions” in old systems**

## Performance Comparison



**The more implementations defined, the higher is the improvement on performance**



Motivation & Overview

Enhancement Framework

Source Code Plugin - Technology

Function Group Enhancement - Technology

Class Enhancement - Technology

Kernel-BAdl – Technology

**Switch Framework**

Summary

## Switch Framework - Motivation

**Goal of Switch Framework:**

**Control visibility of repository objects at runtime through switches**

**The Switch Framework can be used to**

- **Switch on industry solutions / Enterprise Add-ons**
- **Develop new functions without affecting existing ones**
- **Enhance delivered systems at partner and customer site in the context of the enhancement framework with own functions**

**Benefits:**

- **Industry Solutions are available with every release and SP without delay (i.e. timely provision of legal requirements), CRT's\* are no longer necessary for add-on systems**
- **Industry Solutions can be enriched by generic functions from other industries**
- **Synchronization of release cycles and planning**

\* CRT – Conflict Resolution Transport

# Switchable Objects

## Switchable Objects...

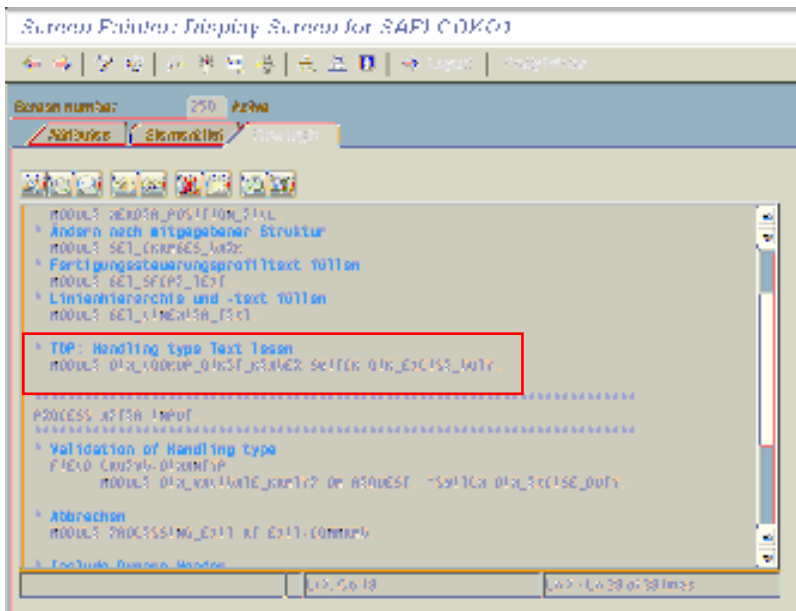
### ...by package assignment

- Appends, SI-, CI-includes for structures in DDIC
- Fixed value appends to domains
- Secondary Indexes
- Append Search Helps
- Enhancement Implementations
- Switch Business Configuration Sets ( Switch BC-Sets)

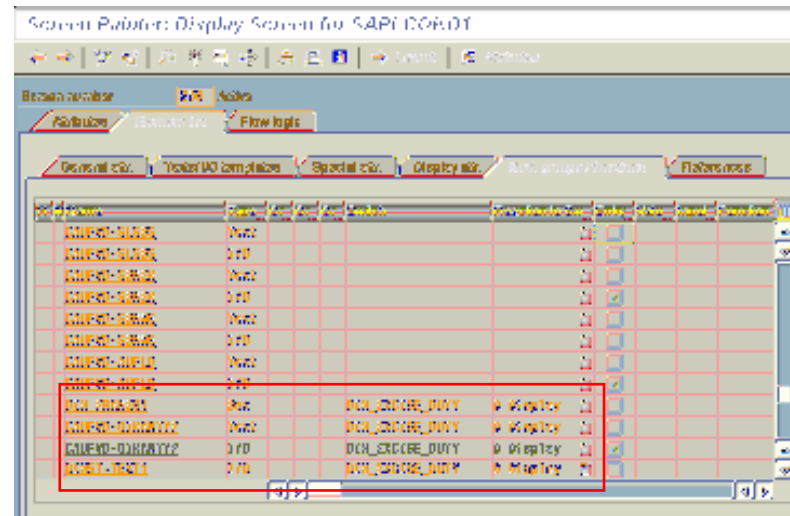
### ...by direct assignment

- Screen elements & Flow logic
- Menu entries & functions
- IMG nodes
- Customizing

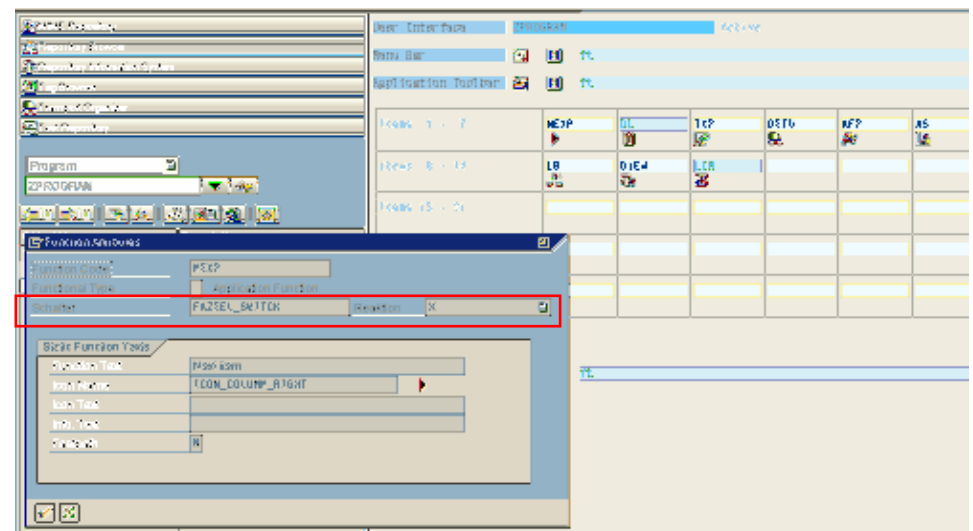
# Direct Assignment: Examples



**Screen Painter: Modules**



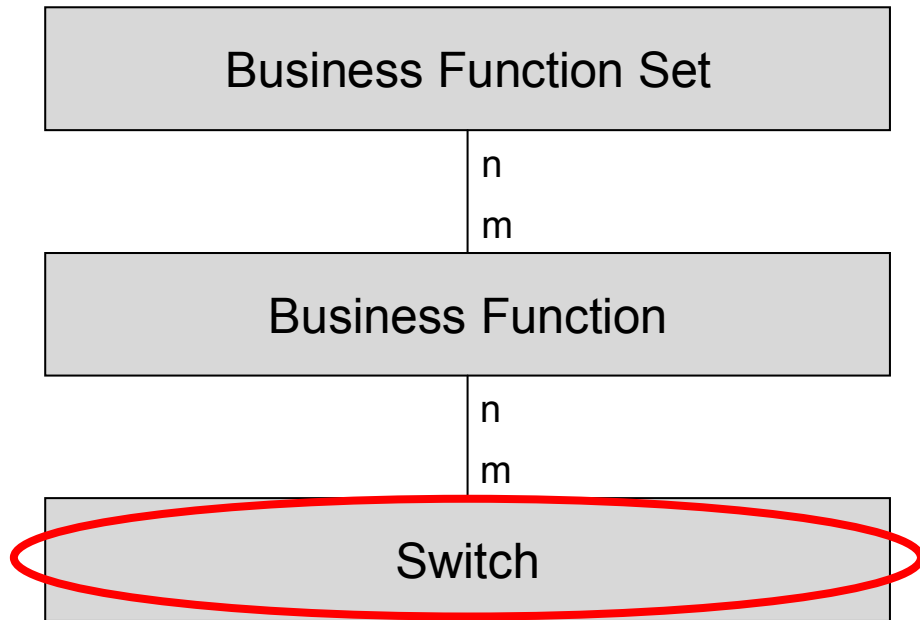
**Screen Painter: Field elements**



**Menu Painter: Function codes**



# Switch



## Switch

- Repository Object
- Calculated states: ON, OFF, STANDBY
- Transaction **SFW1**

Switch Framework

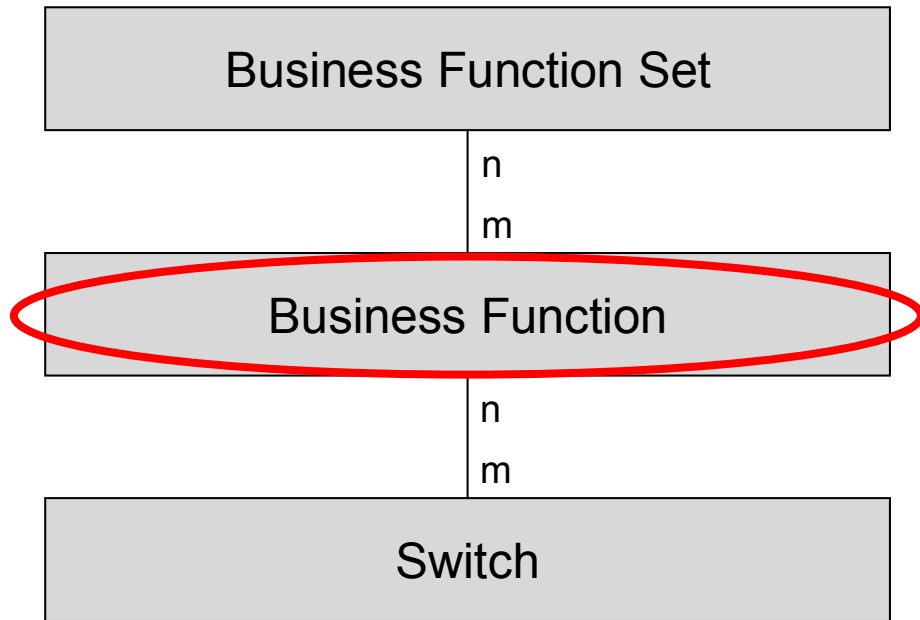
Switch Definition

Switch	Global Status	Short Text	Combi.	DDIC-relevant
DEOMO	off		<input type="checkbox"/>	<input type="checkbox"/>
SEFW_SWITCH_2	on		<input type="checkbox"/>	<input type="checkbox"/>
SEFW_SWITCH_3	on		<input checked="" type="checkbox"/>	<input type="checkbox"/>
SHOWEXTENH	off	Display Implementation	<input type="checkbox"/>	<input type="checkbox"/>
STRAVELAG_APP1	on	STRAVELAG Append 1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
SWITCH_1	off		<input type="checkbox"/>	<input type="checkbox"/>

Packs GUI / Dynpro Area Menu Node MaintViews ViewCluster IMG Node Conf

Switch	Package	Short Description
SEFW_SWITCH_2	SEFW_SWITCHED_2	

# Business Function



## Business Function

- Represents a piece of business functionality
- Contains switches
- Transaction **SFW2**

**Business Function**

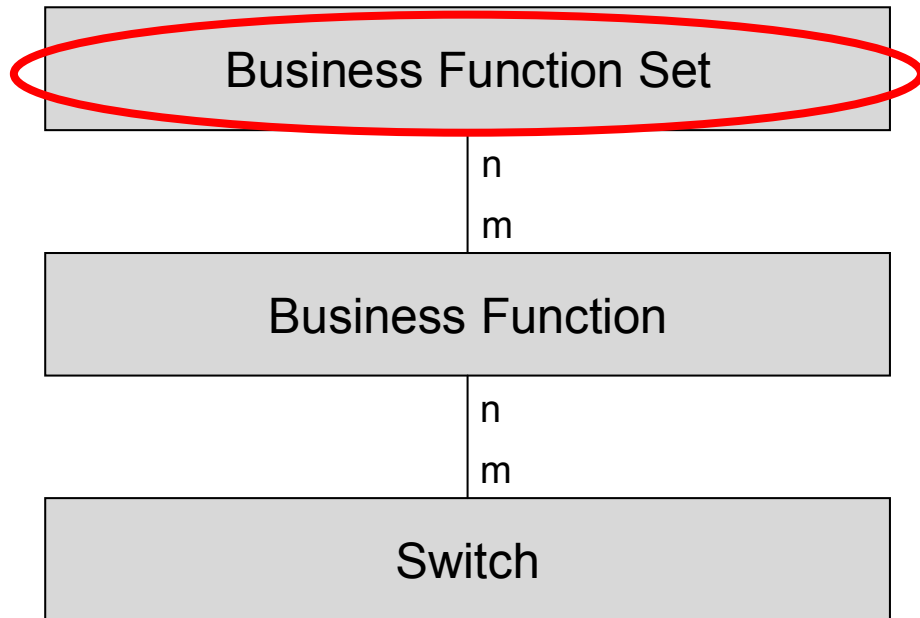
Business Function Definition

Business Function	Short Text	Type	Created By	Created on
BF_1		Industry	DANGERS	13.06.2005
SEFW_BUS_FUNC		Industry	DANGERS	07.06.2005
STRAVELAG_APP1	STRAVELAG Append 1	Industry	DANGERS	02.06.2005

Switches

Business Function	Switch	Type	Short Text
SEFW_BUS_FUNC	SEFW_SWITCH_2	Activation	

# Business Function Set



**Change System Settings**

Syst.Name: RKD  
 Business Function Set: BFS for package SEFW Active

BFS for package SEFW Enterprise AddOns

Business Function	Short Text	Status	Activated on	Active
SEFW_BUS_FUNC		On	16.12.2005 05:26:24	Active
STRAVELAG_APP1	STRAVELAG Append 1	On	16.12.2005 05:26:24	Active

**SFW3**

## Business Function Set

- Pool of business functions
- Represents e.g. one industry solution
- Max. 1 can be active
- Transaction **SFW3** to create BFS
- Use Transaction **SFW5** to activate a BFS.

**Business Function Set**

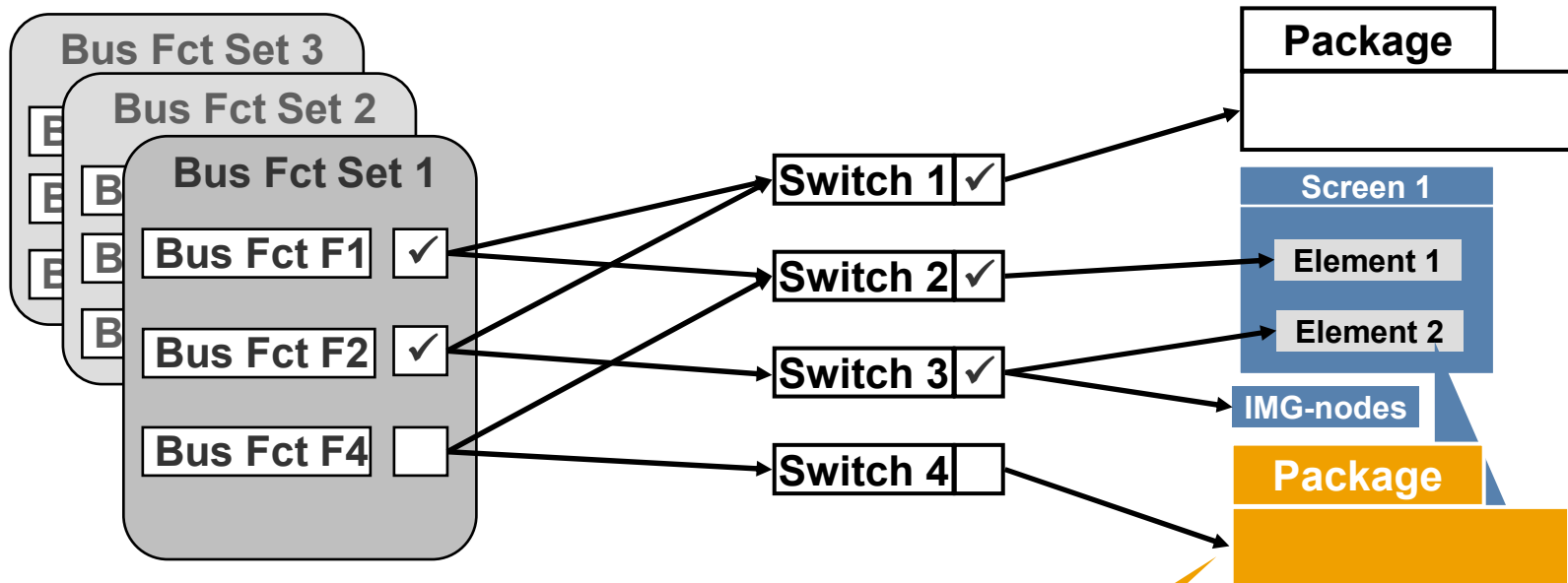
Business Function Set	Short Text	Created By	Created on
SEFW_BFS	BFS for package SEFW	BCKNEIBENK1	30.06.2006

Business Functions

Bus. Func. Set	Business Function	No Display	Always on	Upgrade	Short Text
SEFW_BFS	SEFW_BUS_FUNC	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
SEFW_BFS	STRAVELAG_APP1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	STRAVELAG Append 1

**SFW5**

# Switch Framework: Architecture



Most objects are switched via the assignment of their package to a switch, e.g.

- DDIC objects (e.g. appends)
- Enhancements
- Transactions
- BC-Sets

Objects with no direct package relation are directly assigned to a switch, e.g.

- Screen elements
- Menu entries
- IMG nodes



Motivation & Overview

Enhancement Framework

Source Code Plugin - Technology

Function Group Enhancement - Technology

Class Enhancement - Technology

Kernel-BAdl – Technology

Switch Framework

**Summary**

- The Enhancement Framework offers new possibilities to extend the SAP Standard instead of modifying it.
  - **Source Code** Plugins
  - **Function Group** Enhancements
  - **Class** Enhancements
  - **New BAdIs**
- The new **BAdIs are more flexible and faster** than the classic ones.
- The Enhancements offered by **Enhancement Framework** and some other object types can be **switched by the Switch Framework** as part of a Business Function Set e.g. an industry solution.

## Further Information

### → Help Portal

<http://help.sap.com>

- ➔ Documentation ➔ SAP Netweaver (04s) ➔ Application Platform
- ➔ ABAP technology ➔ ABAP Workbench ➔ Enhancement Framework

### → OKP / RKT Learning Maps

Internal SAP:

<http://intranet.sap.com/rkt-netweaver>

- ➔ Consulting ➔ SAP NW 04s ➔ Creating Business Applications using ABAP

Ramp-up customers:

Send mail to [rkt@sap.com](mailto:rkt@sap.com)

### → SDN

<http://sdn.sap.com>

## Questions?



# Q&A



- No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.
  - Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.
  - Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.
  - IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation.
  - Oracle is a registered trademark of Oracle Corporation.
  - UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.
  - Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.
  - HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.
  - Java is a registered trademark of Sun Microsystems, Inc.
  - JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.
  - MaxDB is a trademark of MySQL AB, Sweden.
  - SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.
- 
- The information in this document is proprietary to SAP. No part of this document may be reproduced, copied, or transmitted in any form or for any purpose without the express prior written permission of SAP AG.
  - This document is a preliminary version and not subject to your license agreement or any other agreement with SAP. This document contains only intended strategies, developments, and functionalities of the SAP® product and is not intended to be binding upon SAP to any particular course of business, product strategy, and/or development. Please note that this document is subject to change and may be changed by SAP at any time without notice.
  - SAP assumes no responsibility for errors or omissions in this document. SAP does not warrant the accuracy or completeness of the information, text, graphics, links, or other items contained within this material. This document is provided without a warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.
  - SAP shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials. This limitation shall not apply in cases of intent or gross negligence.
  - The statutory liability for personal injury and defective products is not affected. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third-party Web pages nor provide any warranty whatsoever relating to third-party Web pages.