

New BAdIs in WCM

Release ERP 6.0, EhP3 + EhP5

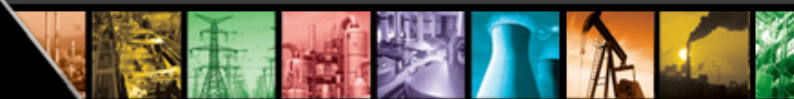


Michael Lesk

WCM Focus Group, April 2010
Tampa, Florida



SAP-CENTRIC EAM
Inspiring Individuals – Empowering Enterprises



wcm

Agenda

- **1. Introduction**
- 2. Process-controlling BAdIs for Order and single WCM Objects (EhP3)
- 3. BAdIs for Additional Data of single WCM Objects (EhP3)
- 4. BAdIs for Screen Enhancements of single WCM Objects (EhP3)
- 5. BAdIs for Menu Enhancements of single WCM Objects (EhP3)
- 6. BAdIs for WCM List Processing Output (EhP5)
- 7. BAdIs for Menu Enhancements of WCM List Processing (EhP5)
- 8. General Design Principles for WCM BAdIs

1) Motivation

■ Situation before Release ERP 6.0, EhP3:

- Not all **individual customer requirements** were already met by standard functionality of Work Clearance Management (WCM).
- That's why customers have enhanced standard functionality individually, which meant a **modification** in most cases.

■ Follow-up action for Release ERP 6.0, EhP3:

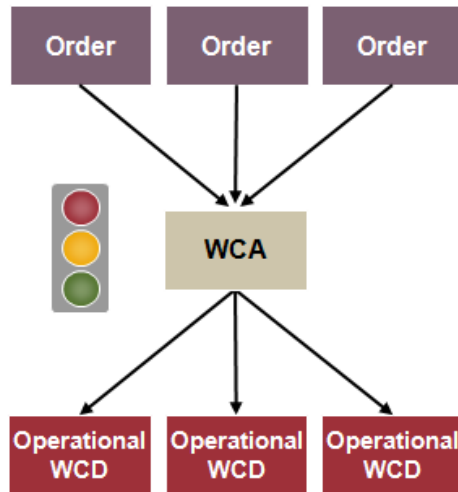
- Based on a roll-in of customer requirements, SAP has provided **various Business-Add-Ins** (BAIs) with release **ERP 6.0, EhP3**, enabling SAP customers to develop individual WCM enhancements **for processing single WCM objects**, without needing to modify the system anymore.

■ Further action for Release ERP 6.0, EhP5:

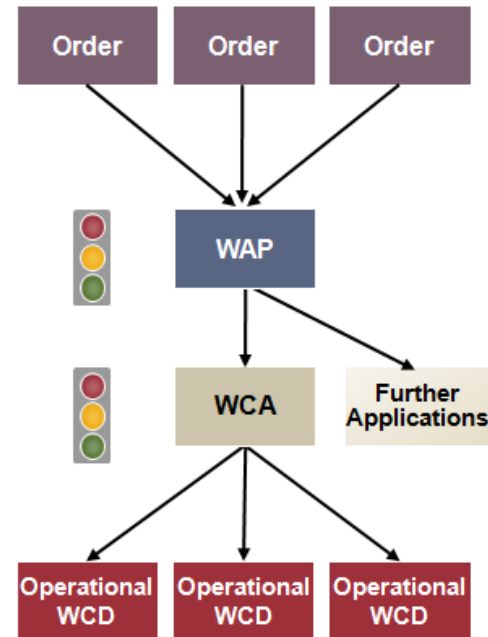
- In addition to the BAIs for processing single WCM objects, SAP provides further BAIs **for list processing of WCM objects** with release **ERP 6.0, EhP5**.



1) Let's start with a look at the WCM Architecture...



The Standard Model



The Enhanced Model

- For the whole WCM process, from the order down to WCM and back, customers asked for the option to influence the process steps according to their individual needs, e.g. by individual business checks.

→ As of EhP3, SAP provides BAdIs for controlling different WCM process steps.

1) ...and continue with a look at a single WCM Object...

WC Application Edit Goto Extras System Help

Change WC Application

Permit Master WP Op. V&B Lists Activity QM

Application 100001 Detailed Insp. Feedwater Pump System 000

Status PREP abc

Valid from 24.06.2009 06:00:00 Valid to 24.06.2009 18:00:00

Priority 4 Low Recall Time 0

Overall Condtn

Revision Phase

Reference Object

Functional loc. BBLAC20 Feedwater Pump System 1

Equipment

Train

Responsibilities Location Data Planning Data

Planner group	100 / 1200	Hr. Reuter
Work Center	MECHANIK / 1200	Mechanical maintenance
Authorizgroup		

Generic object services:
different services are possible

Status processing and audit trail:
system status and user status

Assignment of partners:
different roles possible

Long text processing:
several internal remarks

Assignment of documents:
link to document management system

Assignment of approvals:
cross-document, interactive

Use of catalog technique:
catalog groups and codes

Assignment of technical objects:
equipment and functional locations

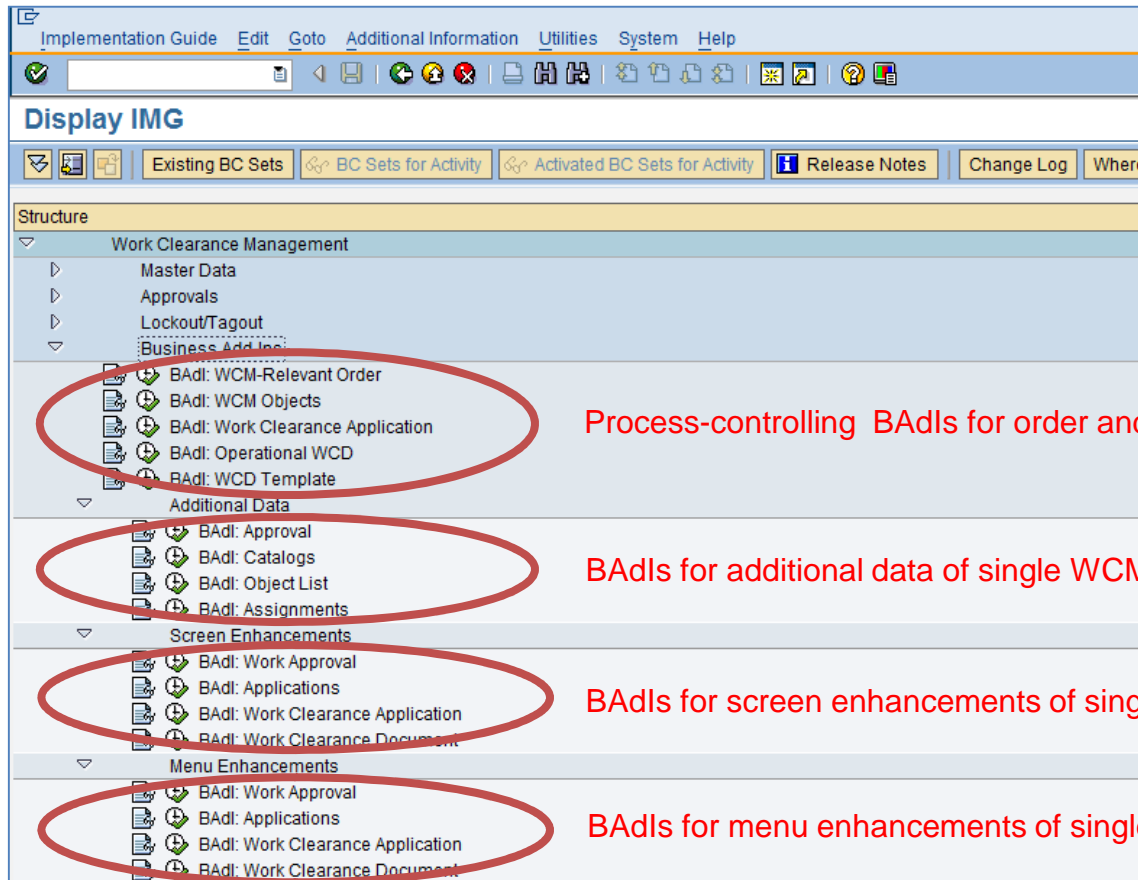
- For processing WCM Objects like e.g. the WCA, SAP was asked for the option to
 - enhance the screens by individual fields → SAP provides screen BAdIs
 - enhance the menus by individual functions → SAP provides menu BAdIs
 - control processing of additional data → SAP provides BAdIs for additional data

1) ...and finish with a look at WCM list processing

WC App.	Short Text	Long Text	Valid From	Valid To
Equipment ObjList Activity QM				
Approvals System Status	UsrStatus	Assgmt	Assgmt	Reference
<input type="checkbox"/> 100000016500	Check Water Supply		25.06.2009	25.06.2010
<input type="checkbox"/> 100000016900	Repair Grime Blower Layer 1		20.07.2009	20.07.2009
<input type="checkbox"/> 100000017000	Repair Compressed Air-Supply		20.07.2009	20.07.2009
<input type="checkbox"/> 100000017400	Repair Grime Blower Layer 3		18.11.2009	18.11.2010
<input type="checkbox"/> 100000016301	Repair Compressed Air-Supply		21.06.2009	21.06.2010
ML-3000				

- For list processing of WCM Objects like e.g. WCAs, SAP was asked for the option to
 - add individual fields to the output list → SAP provides BAdIs for list output
 - enhance the menus by individual functions → SAP provides menu BAdIs for lists

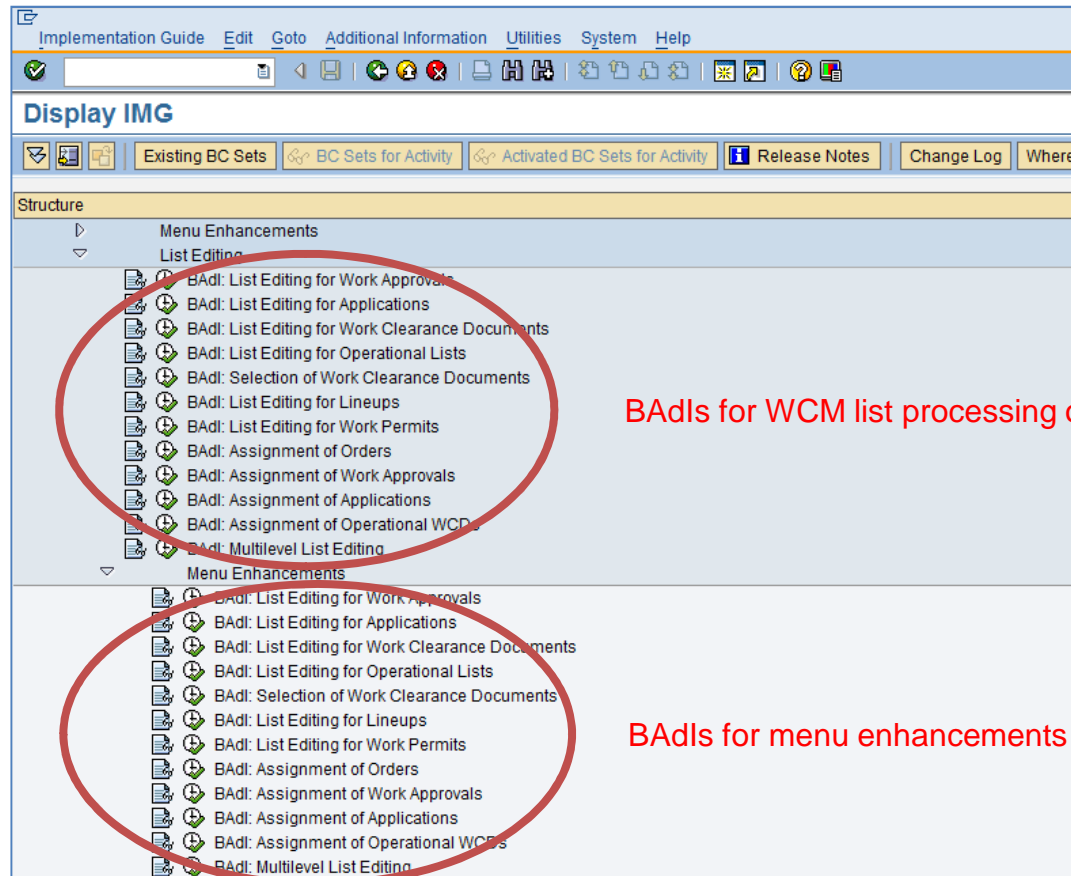
1) Let's have a first look at the BAdIs available in IMG...



- Each BAdI interface for single WCM objects can be accessed via BAdI Builder, either explicitly **via SE18** or implicitly **via above IMG activities**.
 - Note that implicit access via IMG requires activation of the EAM business function **LOG_EAM_CI_2** in the Switch Framework.



1) Let's have a second look at the BAdIs available in IMG...



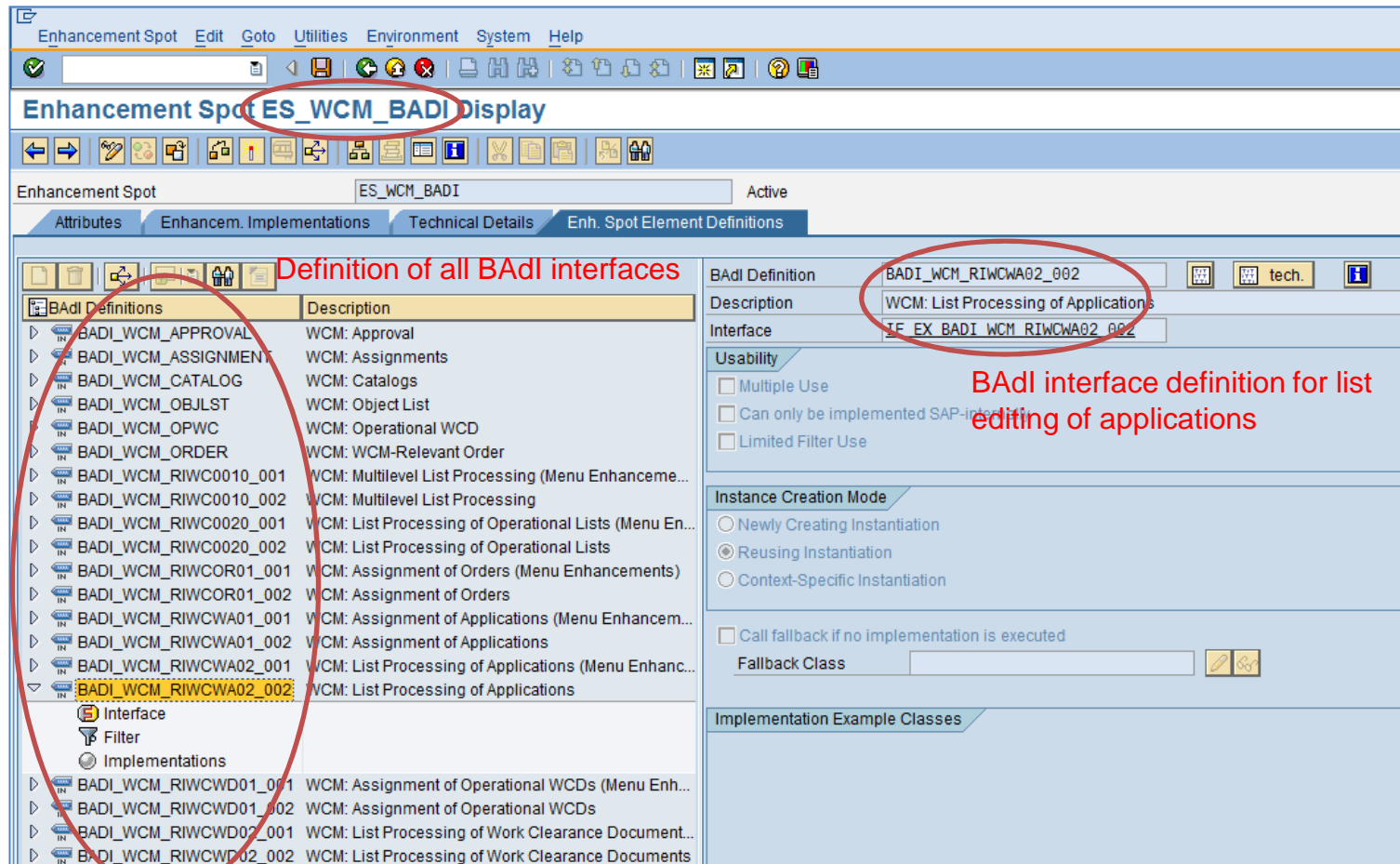
BAdIs for WCM list processing output (EhP5)

BAdIs for menu enhancements of WCM list processing (EhP5)

- Each BAdI interface for list processing of WCM objects can be accessed via BAdI Builder, again either explicitly **via SE18** or implicitly **via above IMG activities**.
 - Note that implicit access via IMG requires activation of the EAM business function **LOG_EAM_WCM_1** in the Switch Framework.

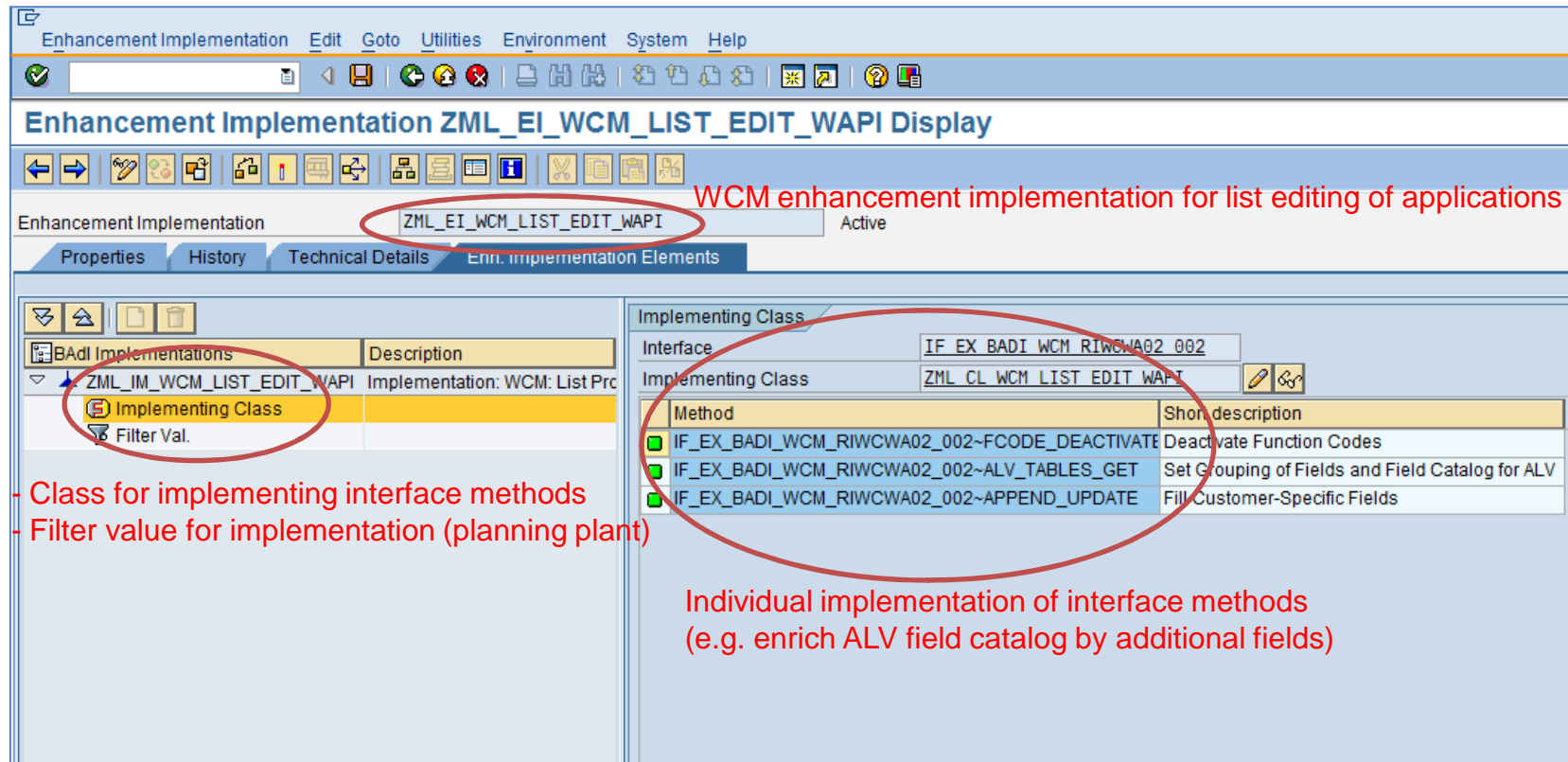


1) ...and continue with a look at the BAdI Builder...



- Each WCM BAdI interface belongs to the same WCM enhancement spot, called **ES_WCM_BADI**.

1) ...and navigate from Interface down to Implementation



The screenshot shows the SAP Enhancement Implementation ZML_EI_WCM_LIST_EDIT_WAPI Display. The title bar indicates the current object is ZML_EI_WCM_LIST_EDIT_WAPI, which is active. The main area is divided into two panes. The left pane, titled 'BADl Implementations', shows a tree structure with 'ZML_IM_WCM_LIST_EDIT_WAPI' selected, which is an 'Implementing Class'. The right pane, titled 'Implementing Class', shows the 'Interface' as 'IF_EX_BADI_WCM_RIWCWA02_002' and the 'Implementing Class' as 'ZML_CL_WCM_LIST_EDIT_WAPI'. Below this, a table lists the methods implemented by the class:

Method	Short description
IF_EX_BADI_WCM_RIWCWA02_002~FCODE_DEACTIVATE	Deactivate Function Codes
IF_EX_BADI_WCM_RIWCWA02_002~ALV_TABLES_GET	Set Grouping of Fields and Field Catalog for ALV
IF_EX_BADI_WCM_RIWCWA02_002~APPEND_UPDATE	Fill Customer-Specific Fields

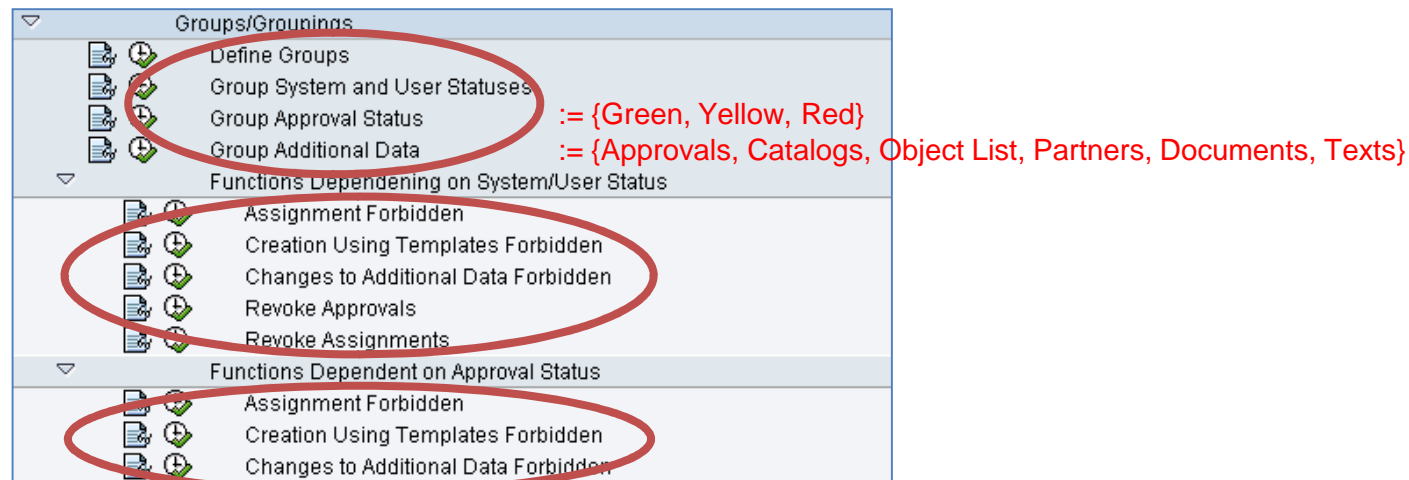
Red annotations highlight key elements: the title bar text 'WCM enhancement implementation for list editing of applications', the 'Implementing Class' label in the left pane, the 'Filter Val.' option, and the table of methods in the right pane. A note explains that the table shows the 'Individual implementation of interface methods (e.g. enrich ALV field catalog by additional fields)'.

- A WCM enhancement implementation may contain several implementing classes, e.g. for different planning plants.
- However, there should be a clear relation between interface and implementing class: Each implementing class should implement exactly one interface!

1) Before having a closer look at the WCM BAdIs...

- ... we want to note the very powerful concept of **status groups**:
 - Pre-defined influence options on certain process steps, based on
 - Approval status (as of release 4.7)
 - System status (as of release 4.7)
 - User status (as of release ERP 6.0, EhP4)
 - Enabled via Customizing, no additional implementation required.

■ IMG:



- Examples (for functions dependent on system status):
 - No partner assignment as long as final safety approval in WCA is not issued.
 - Revoke final safety approval in WCA when test cycle is triggered.



Agenda

- 1. Introduction
- **2. Process-controlling BAdIs for Order and single WCM Objects (EhP3)**
- 3. BAdIs for Additional Data of single WCM Objects (EhP3)
- 4. BAdIs for Screen Enhancements of single WCM Objects (EhP3)
- 5. BAdIs for Menu Enhancements of single WCM Objects (EhP3)
- 6. BAdIs for WCM List Processing Output (EhP5)
- 7. BAdIs for Menu Enhancements of WCM List Processing (EhP5)
- 8. General Design Principles for WCM BAdIs

2) Process-controlling BAdIs for Order and single WCM Objects

- In each case one BAdI

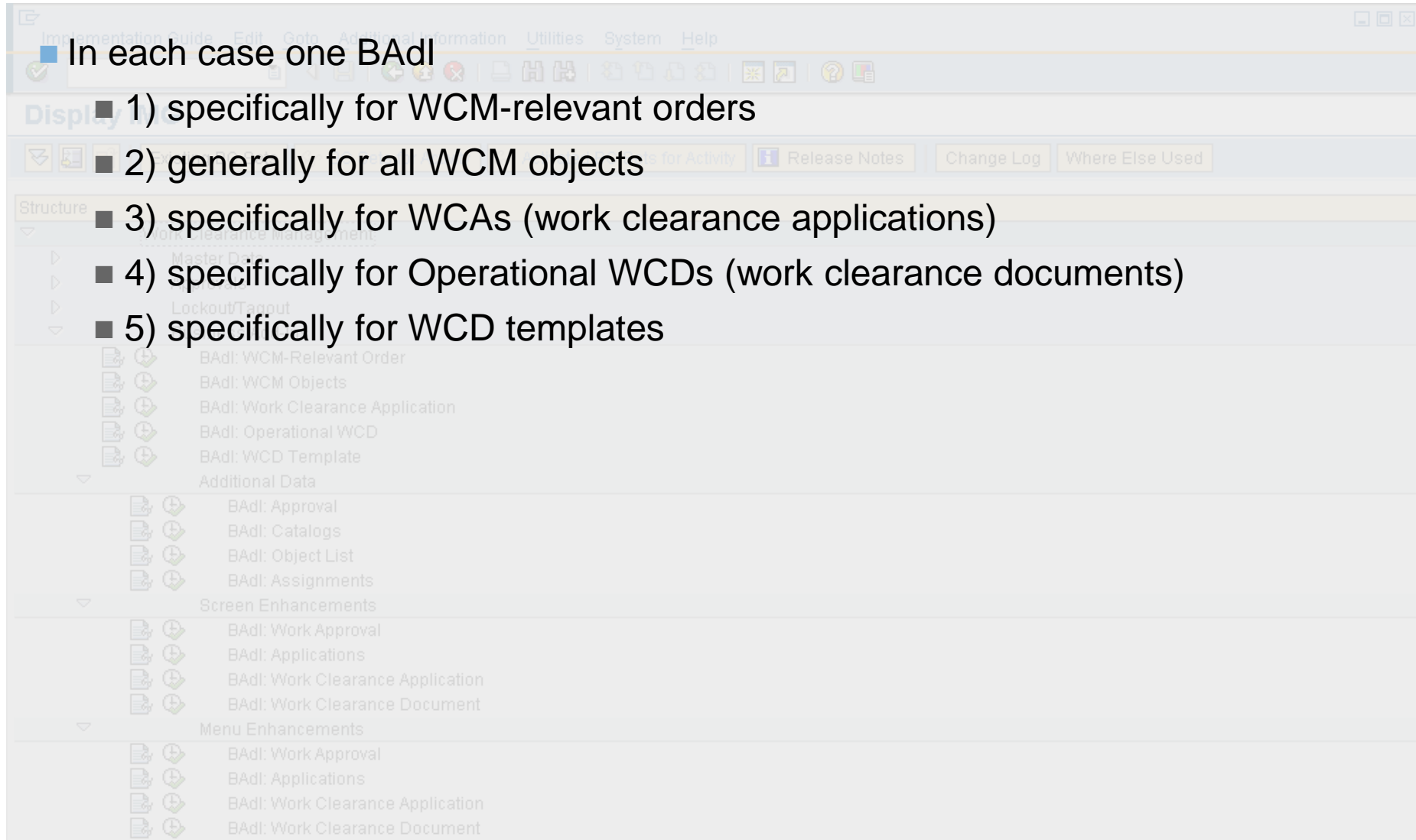
- 1) specifically for WCM-relevant orders

- 2) generally for all WCM objects

- 3) specifically for WCAs (work clearance applications)

- 4) specifically for Operational WCDs (work clearance documents)

- 5) specifically for WCD templates



2.1) Process-controlling BAdI for WCM-relevant Orders Header Level

- General BAdI methods for WCM-relevant orders:
 - Change color or short text (e.g. in multi-level list)

- Methods for valuation of WCM-relevant orders:
 - Deactivate function codes in the valuation dialog
 - Check if application can be valued as requested (“Yes”, “No”)
 - Check if the valuation dialog can be closed as requested (by the green check)

- Methods for work release/completion in a WCM-relevant order:
 - Check if “Release for Execution” can be issued as requested
 - Check if “Work Completed” can be confirmed as requested

2.1) Process-controlling BAdI for WCM-relevant Orders Operation Level

- Methods for maintaining WCM-relevant operations:
 - Deactivate function codes in the operation dialog
 - Check if approval can be valuated as requested (“Yes”, “No”)
 - Check if the operation dialog can be closed as requested (by the green check)

- Methods for work release/completion of WCM-relevant operations:
 - Check if “Release for Execution” can be issued as requested
 - Check if “Work Completed” can be confirmed as requested

2.1) Example

Check if Release for Execution can be issued as requested (1/2)

Order Edit Goto Extras Environment System Help

Change Maintenance Order 833026: Ready for execution from standard WCM perspective

WC Application Released for execution CET

Order PM01 833026 Repair of Pump Station Notification

System Status WCM CRTD MANC NMAT PRC

PMActType 103 Repair

Order Data 4 Order Data 5 Order Data 6

Responsibilities

Planner Group 100 / 1000 Hr. Weber

Main WorkCtr MECHANIK / 1000 Mechanical maintainer

Employee Responsib

Reference Object

Functional Loc. 01-B01 Pump station

Equipment

Assembly

Dates

Bas. start date 01.04.2009 Start Time 16:09:00 Priority

Basic fin. date 30.04.2009 Basic Fin. Time 16:09:00 Revision

Basic duration of work

2.1) Example

Check if Release for Execution can be issued as requested (2/2)

Order Edit Goto Extras Environment System Help

Change Maintenance Order 833026: Request "Release for execution"

WC Application Released for execution CET

Order PM01 833026 Repair of Pump Station Notification

System Status WCM CRTD MANC NMAT PRC

PMActType 103 Repair

Order Data 4 Order Data 5 Order Data 6

Responsibilities

Planner Group 100 / 1000 Hr. Weber

Main WorkCtr MECHANIK / 1000 Mechanical maintainer

Employee Responsit

Reference Object

Functional Loc. 01 - B01

Equipment

Assembly

Dates

Bas. start date 01.04.2009 Start Time 16:09:00 Priority

Basic fin. date 30.04.2009 Basic Fin. Time 16:09:00 Revision

Information

BAAd: Additional check for EXEC: Order not covered by WC validity

Check result when requesting "Release for execution"

2.2) Process-controlling BAdI for all WCM Objects

General Functions

- General BAdI methods for WCM objects:
 - Change color or short text (e.g. in multi-level list)
 - Set user field for influencing standard field selection
 - Set change indicator for WCM object (e.g. for confirmation prompt upon exit)
 - Check WCM object in case of requested save

- Methods for basic functions of WCM objects:
 - Check if copy template can be used when creating WCM object (with template)
 - Set planning data of WCM object (start/end of basic, scheduled & actual dates)
 - Deactivate function codes for the WCM object being processed
 - Check requested print of WCM object header data
 - Check requested print of work permit

2.2) Process-controlling BAdI for all WCM Objects Status Changes

- Methods for status changes of WCM objects:
 - Check if the status of the WCM object can be changed as requested. The following status changes can be checked:
 - Preparation and change mode
 - Completion and rejection
 - Set and reset inactivation flag
 - Set and reset deletion flag

Check if Operational WCD can be prepared as requested (2/2)



2.3) Process-controlling BAdI for WCAs

- Methods for the test cycle controlled by WCAs:
 - Check if test cycle can be permitted as requested
 - Check if existing permit for test cycle can be revoked as requested
 - Rename the functions permitting a test cycle and revoking the permit

2.4) Process-controlling BAdI for Operational WCDs

General Functions

- General methods for Operational WCD items:
 - Change logical destination of RFC server connecting a graphical system
 - Revise data imported from graphical system
 - Revise technical objects selected via multiple selection
 - Input help and input check for lock (physical blocking)
 - Set user field for influencing standard field selection
 - Default sort sequence of the items of an Operational WCD on the maintenance screen as well as on the switching screen
 - Extend the standard simulation on the maintenance screen (→ internal checks) and on the switching screen (→ external checks) by individual checks
 - Rename the functions (pushbuttons, tooltips) within the operational cycle

- Methods for basic functions of Operational WCDs:
 - Check requested print of a tagging list or an untagging list for selected items
 - Check requested print of tags or test tags for selected items
 - Check if the WCD is untaggable

2.4) Process-controlling BAdI for Operational WCDs Status Changes

- Methods for status changes of Operational WCD items:
 - Check if the status of a WCD item can be changed as requested. The following status changes can be checked:
 - Inactivation
 - Set and reset of operational protection
 - Setting of all (applicable) status of the operational cycle, i.e.:
 - “Tag” (:= to be tagged)
 - “Tag Printed”
 - “Tagged”
 - “Untag Temporarily” (:= to be temporarily untagged)
 - “Test Tag Printed”
 - “Temporarily Untagged”
 - “Untag” (:= to be untagged)
 - “Untagged”

2.5) Process-controlling BAdI for WCD Templates

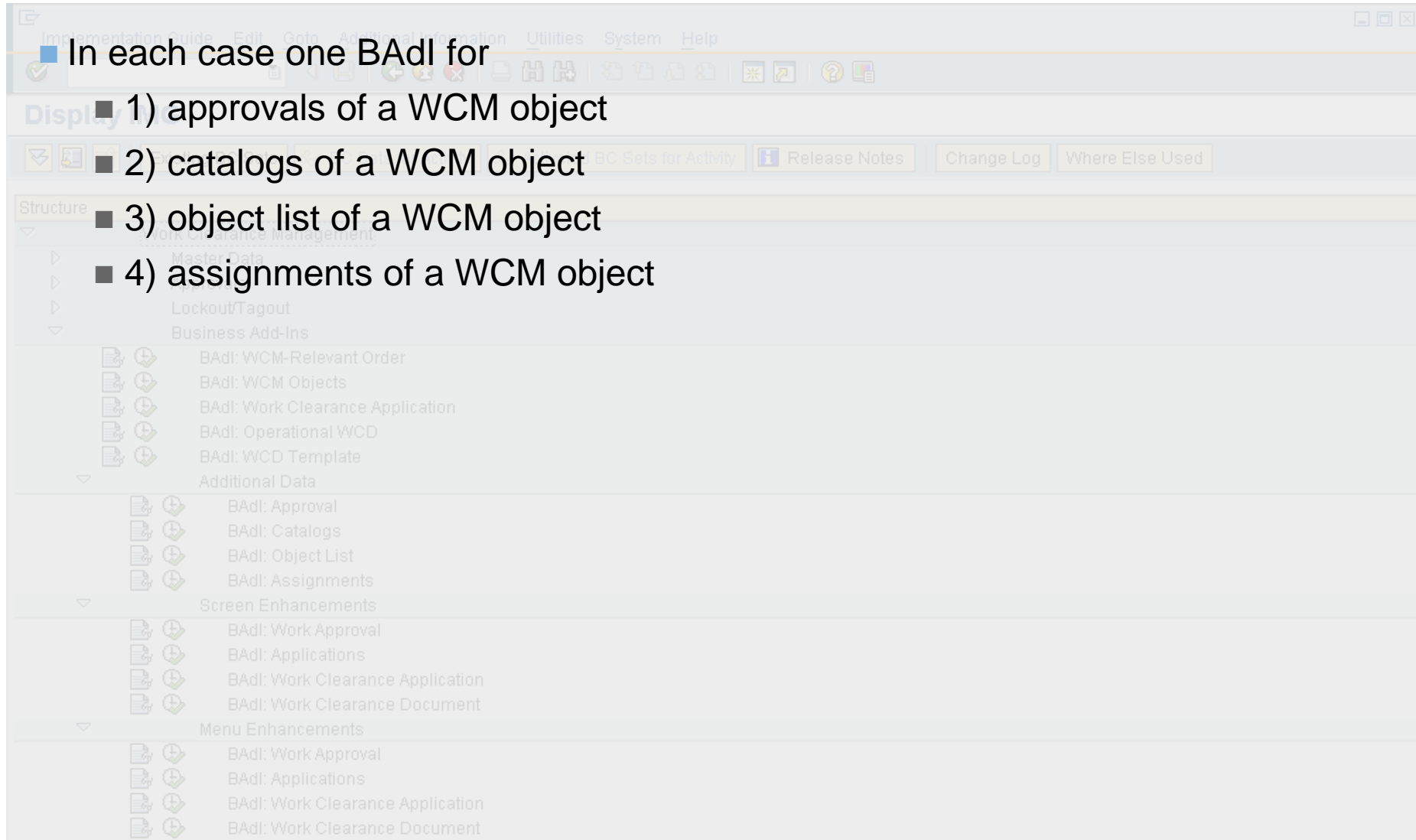
- Methods for WCD templates:
 - Default sort sequence of the items of a WCD template on the maintenance screen
 - Extend the standard simulation on the maintenance screen by individual checks
 - Check if the status of a WCD item can be set to “Inactive” as requested

Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and single WCM Objects (EhP3)
- **3. BAdIs for Additional Data of single WCM Objects (EhP3)**
- 4. BAdIs for Screen Enhancements of single WCM Objects (EhP3)
- 5. BAdIs for Menu Enhancements of single WCM Objects (EhP3)
- 6. BAdIs for WCM List Processing Output (EhP5)
- 7. BAdIs for Menu Enhancements of WCM List Processing (EhP5)
- 8. General Design Principles for WCM BAdIs

3) BAdIs for additional data of single WCM Objects

- In each case one BAdI for
 - 1) approvals of a WCM object
 - 2) catalogs of a WCM object
 - 3) object list of a WCM object
 - 4) assignments of a WCM object



3.1) BAdI for approvals of a WCM Object

- The BAdI provides the following implementation options:
 - Deactivate functions for approval assignment
 - Check if the assignment of an approval can be removed as requested
 - Check if the assigned approval can be issued as requested
 - Check if the issue of an assigned approval can be revoked as requested
 - Check if the dialog window for approval assignment can be closed as requested

3.2) BAdI for Catalogs of a WCM Object

- The BAdI provides the following implementation options:
 - Deactivate functions for catalog maintenance
 - Default sort sequence for the maintained entries of the assigned catalog
 - Check if a maintained entry for the assigned catalog can be removed as requested
 - Check if a maintained entry for the assigned catalog can be valued as requested
 - Check if the dialog window for catalog maintenance can be closed as requested

3.3) BAdI for Object List of a WCM Object

- The BAdI provides the following implementation options:
 - Deactivate functions for object list maintenance
 - Default sort sequence for the maintained entries of the object list
 - Check if a maintained entry of the object list can be removed as requested
 - Check if the dialog window for object list maintenance can be closed as requested

3.4) BAdI for (superior/subordinate) Assignments of a WCM Object

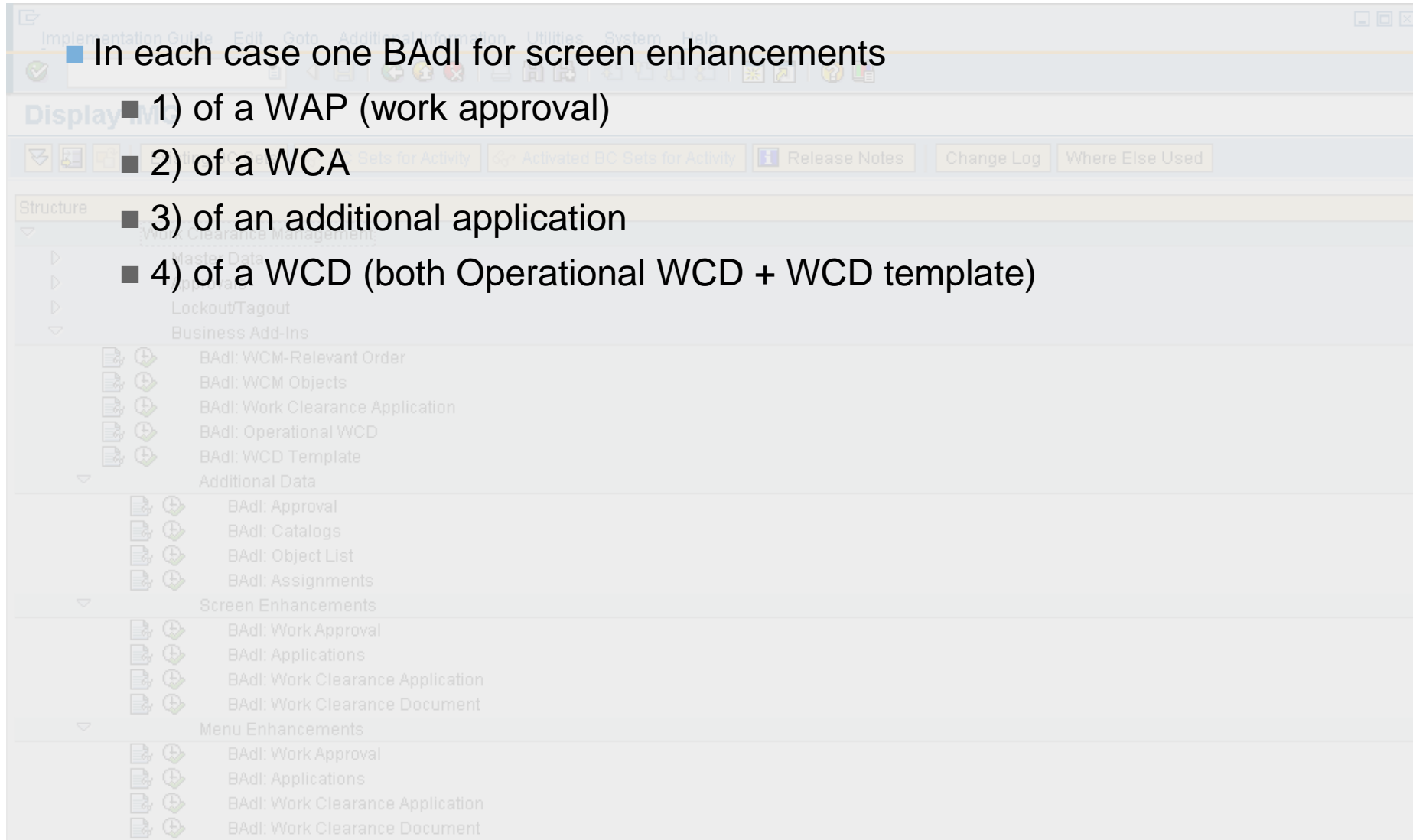
- The BAdI provides the following implementation options:
 - Deactivate functions for maintaining assignments
 - Check if a maintained assignment can be removed as requested
 - Check if the dialog window for maintaining superior and subordinate assignments can be closed as requested
 - Restrict the set of applicable objects for an assignment
 - Example (Enhanced Model): Restriction of work approvals when assignment is requested during maintenance of an order (subordinate assignment) or of an application (superior assignment)

Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and single WCM Objects (EhP3)
- 3. BAdIs for Additional Data of single WCM Objects (EhP3)
- **4. BAdIs for Screen Enhancements of single WCM Objects (EhP3)**
- 5. BAdIs for Menu Enhancements of single WCM Objects (EhP3)
- 6. BAdIs for WCM List Processing Output (EhP5)
- 7. BAdIs for Menu Enhancements of WCM List Processing (EhP5)
- 8. General Design Principles for WCM BAdIs

4) BAdIs for Screen Enhancements of single WCM Objects

- In each case one BAdI for screen enhancements
 - 1) of a WAP (work approval)
 - 2) of a WCA
 - 3) of an additional application
 - 4) of a WCD (both Operational WCD + WCD template)



4) BAdIs for Screen Enhancements per WCM Object

- Screen BAdIs are built identically for all WCM objects. They consist of **two basic parts**:
 - The **screen area** provided for the screen enhancement of a WCM object.
 - In **EhP3 + EhP4**, SAP provides **one** screen enhancement area, integrated as **4th tab** (next to Responsibilities, Location Data, Planning Data) on the header screen of a WCM object. On this tab, one customer-specific include subscreen is embedded.
 - As of **EhP5**, view profiles for WCM objects will support a **flexible arrangement** of all header subscreens, including **two** screen enhancement areas, each of them containing a customer-specific include subscreen.
 - Furthermore, screen enhancement of a WCM object requires implementing the following **interface methods** of the underlying BAdI:
 - SUBSCREEN_DATA_GET: Data communication from the customer-specific include screen to the outside (→ PAI)
 - SUBSCREEN_DATA_SET: Data communication from outside to the customer-specific include screen (→ PBO)
 - TAB_PAGE_TITLE_GET: Set title for the 4th tab on the header screen



4) Screen Enhancement Implementation: Screen Area

Enhancement Implementation ZML_EI_WP_SCRN_WAPI Display

Enhancement Implementation: ZML_EI_WP_SCRN_WAPI Active

Properties History Technical Details Enh. Implementation Elements

BAOI Implementations Description

- ZML_IM_WP_SCRN_V Implementation: WCM: R...
- Implementing Clas
- Filter Val.
- Screen Enhancement

Screen Enhancements

BAOI Definition	BADI_WCM_WAPI_001			
BAOI Implementation	ZML_IM_WP_SCRN_WAPI			
Tcode/program	Scr...	Subscreen Area	Program	Sub...
SAPLWCFI	8300	USER1	SAPLZML_WP	10
SAPLWCFI	8400	USER2	SAPLZML_WP	20

Standard include subscreens: Customer-specific data subscreens

- 8300 (as of EhP3)
- 8400 (as of EhP5)

4) Screen Enhancement Implementation: Class Interface

The screenshot displays the SAP Enhancement Implementation Workbench interface. The title bar reads "Enhancement Implementation ZML_EI_WP_SCRN_WAPI Display". The main window has a menu bar (Enhancement Implementation, Edit, Goto, Utilities, Environment, System, Help) and a toolbar. Below the toolbar, the "Enhancement Implementation" section shows the object "ZML_EI_WP_SCRN_WAPI" as "Active". The "Enh. Implementation Elements" tab is selected, showing a tree view on the left with "Implementing Class" highlighted. The right pane shows the "Implementing Class" details for the interface "IF_EX_BADI_WCM_WAPI_001" and the implementing class "ZML_CL_WP_SCRN_WAPI". A table lists three methods, all of which are circled in red:

Method	Short description
IF_EX_BADI_WCM_WAPI_001~SUBSCREEN_DATA_SET	Transport Data from Subscreen
IF_EX_BADI_WCM_WAPI_001~SUBSCREEN_DATA_SET	Transport Data to Subscreen
IF_EX_BADI_WCM_WAPI_001~TAB_PAGE_TITLE_GET	Set Tab Page Title

Individual class implementation (three methods)

Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and single WCM Objects (EhP3)
- 3. BAdIs for Additional Data of single WCM Objects (EhP3)
- 4. BAdIs for Screen Enhancements of single WCM Objects (EhP3)
- **5. BAdIs for Menu Enhancements of single WCM Objects (EhP3)**
- 6. BAdIs for WCM List Processing Output (EhP5)
- 7. BAdIs for Menu Enhancements of WCM List Processing (EhP5)
- 8. General Design Principles for WCM BAdIs

5) BAdIs for Menu Enhancements of single WCM Objects

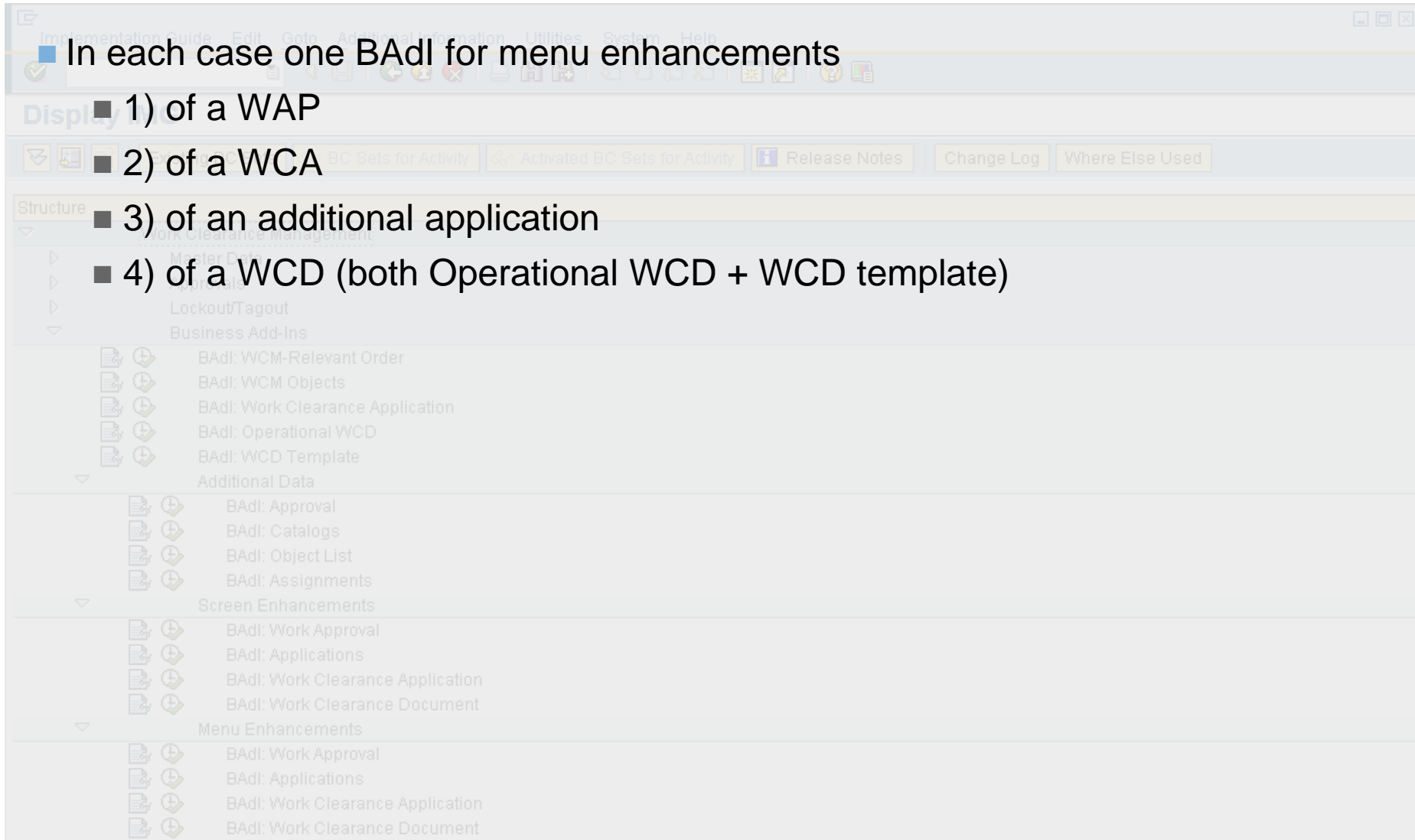
■ In each case one BAdI for menu enhancements

■ 1) of a WAP

■ 2) of a WCA

■ 3) of an additional application

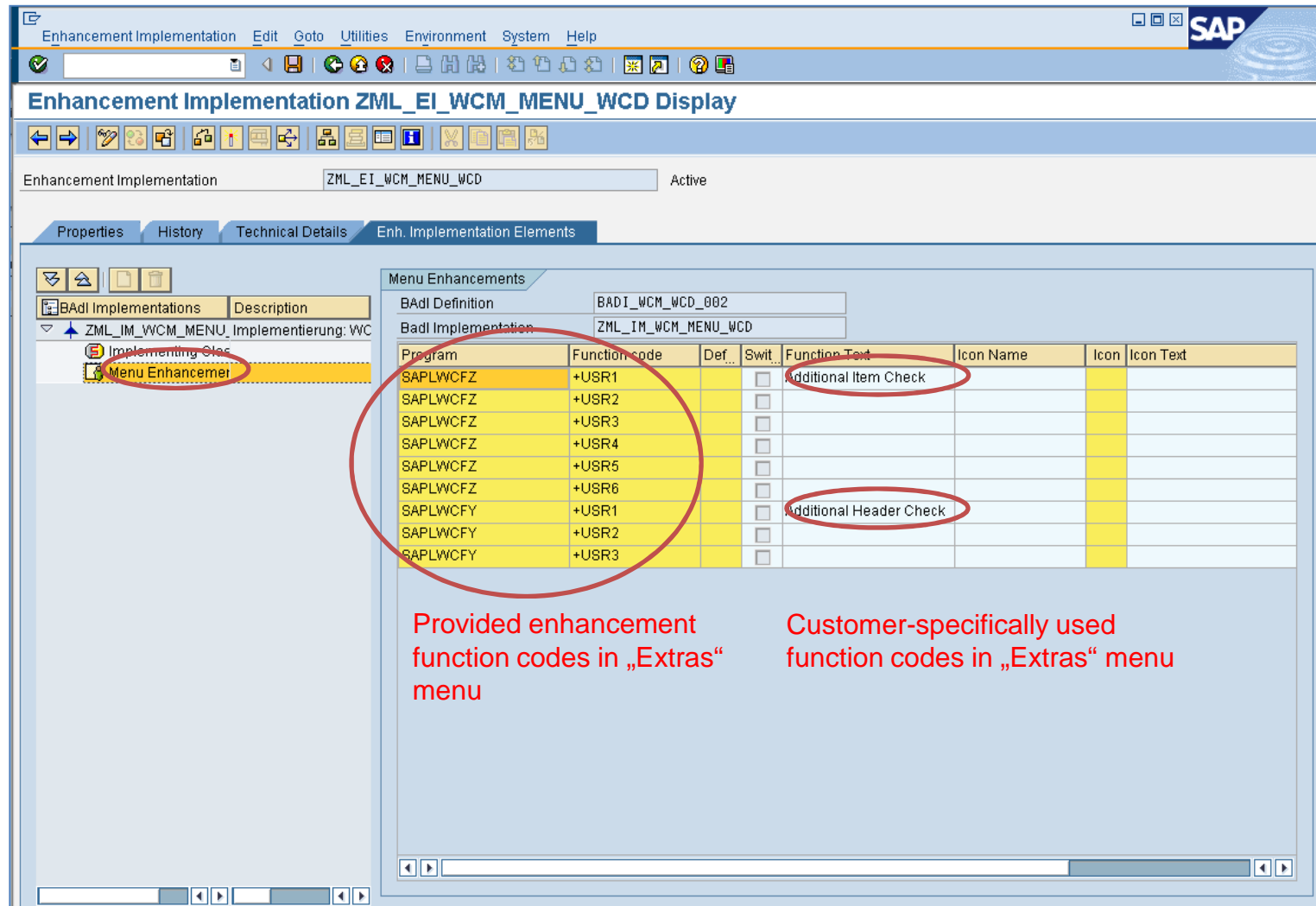
■ 4) of a WCD (both Operational WCD + WCD template)



5) BAdIs for Menu Enhancements per WCM Object

- Menu BAdIs are built identically for all WCM objects. They consist of **two basic parts**:
 - The **function codes** provided for the menu enhancement of a WCM object are integrated in the “Extras” menu on the header screen of a WCM object as well as on its item screens (→ maintenance screen, switching screen).
 - On the **header** screen it is possible to define **up to three** customer-specific function codes.
 - On the **item** screens it is possible to define **up to six** customer-specific function codes.
 - Furthermore, menu enhancement of a WCM object requires implementing the following **interface method** of the underlying BAdI:
 - FCODE_EXECUTE: Execute a customer-specific function code

5) Menu Enhancement Implementation: Function Codes



The screenshot shows the SAP Enhancement Implementation ZML_EI_WCM_MENU_WCD Display. The interface includes a menu bar at the top with options like Enhancement Implementation, Edit, Goto, Utilities, Environment, System, and Help. Below the menu bar is a toolbar with various icons. The main area is divided into several sections. On the left, there is a tree view showing the hierarchy of enhancements, with 'Menu Enhancements' selected. The main area displays the 'Menu Enhancements' table, which lists various function codes and their descriptions. The table has columns for Program, Function code, Def..., Swit..., Function Text, Icon Name, Icon, and Icon Text. The function codes are listed in two groups: 'Additional Item Check' and 'Additional Header Check'. The 'Additional Item Check' group includes function codes +USR1 through +USR6, all with the program SAPLWCFZ. The 'Additional Header Check' group includes function codes +USR1 through +USR3, with the program SAPLWCFY. The 'Additional Item Check' group is circled in red, and the 'Additional Header Check' group is also circled in red. Below the table, there are two red text annotations: 'Provided enhancement function codes in „Extras“ menu' and 'Customer-specifically used function codes in „Extras“ menu'.

Program	Function code	Def...	Swit...	Function Text	Icon Name	Icon	Icon Text
SAPLWCFZ	+USR1		<input type="checkbox"/>	Additional Item Check			
SAPLWCFZ	+USR2		<input type="checkbox"/>				
SAPLWCFZ	+USR3		<input type="checkbox"/>				
SAPLWCFZ	+USR4		<input type="checkbox"/>				
SAPLWCFZ	+USR5		<input type="checkbox"/>				
SAPLWCFZ	+USR6		<input type="checkbox"/>				
SAPLWCFY	+USR1		<input type="checkbox"/>	Additional Header Check			
SAPLWCFY	+USR2		<input type="checkbox"/>				
SAPLWCFY	+USR3		<input type="checkbox"/>				

Provided enhancement function codes in „Extras“ menu

Customer-specifically used function codes in „Extras“ menu

5) Menu Enhancement Implementation: Class Interface

Enhancement Implementation ZML_EI_WCM_MENU_WCD Display

Enhancement Implementation: ZML_EI_WCM_MENU_WCD Active

Properties History Technical Details Enh. Implementation Elements

Implementing Class

Interface	Implementing Class
IF_EX_BADI_WCM_WCD_002	ZML_CL_WCM_MENU_WCD

Method	Short description
IF_EX_BADI_WCM_WCD_002~FCODE_EXECUTE	Execute Function Code

Individual class implementation (method FCODE_EXECUTE)

Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and single WCM Objects (EhP3)
- 3. BAdIs for Additional Data of single WCM Objects (EhP3)
- 4. BAdIs for Screen Enhancements of single WCM Objects (EhP3)
- 5. BAdIs for Menu Enhancements of single WCM Objects (EhP3)
- **6. BAdIs for WCM List Processing Output (EhP5)**
- 7. BAdIs for Menu Enhancements of WCM List Processing (EhP5)
- 8. General Design Principles for WCM BAdIs

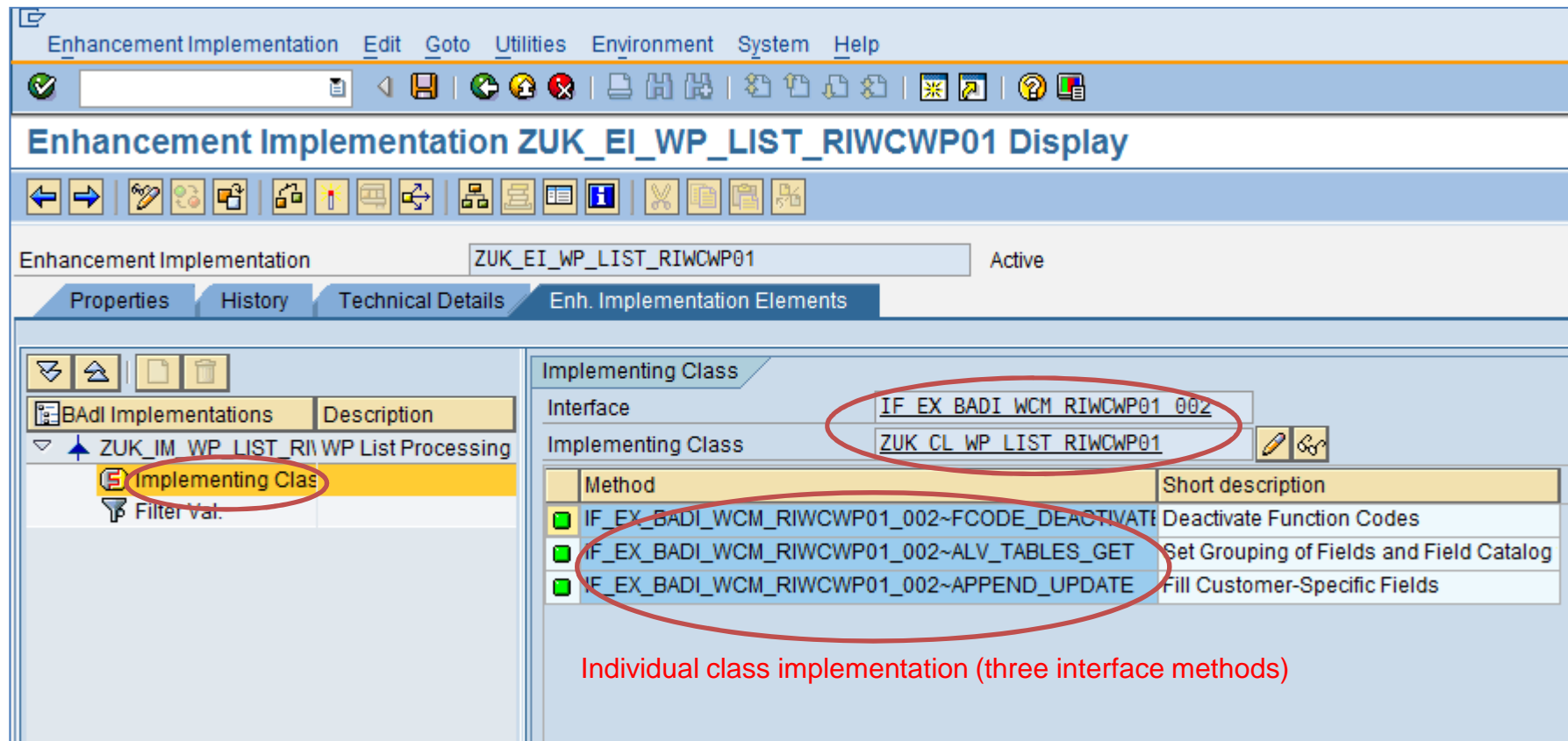
6) BAdIs for WCM List Processing Output

- In each case one BAdI for **native list processing** output
 - 1) of WAPs
 - 2) of applications
 - 3) of WCDs (both Operational WCD + WCD template)
 - 4) of operational lists
 - 5) of lineups
- In each case one BAdI for list selection output **during assignment**
 - 1) of orders
 - 2) of WAPs
 - 3) of applications
 - 4) of Operational WCDs
- One BAdI for WCD list selection output **within clipboard**
- One BAdI for the combined **work permit/order list** processing output
- One BAdI for **multilevel list** processing output

6) BAdIs for WCM List Processing Output

- BAdIs for the list processing output are built (nearly) identical for all WCM objects:
 - All BAdIs (except the one for the multilevel list) provide the following interface methods:
 - ALV_TABLES_GET: Set ALV field catalog and grouping of fields for the output list
 - APPEND_UPDATE: Fill customer-specific fields in the output list
 - FCODE_DEACTIVATE: Deactivate function codes for the output list
 - The BAdI for the multilevel list provides the following interface method:
 - FCODE_DEACTIVATE: Deactivate function codes for the output list

6) BAdI Implementation for List Processing Output: Class Interface



The screenshot shows the SAP Enhancement Implementation interface for ZUK_EI_WP_LIST_RIWCWP01. The main window is titled "Enhancement Implementation ZUK_EI_WP_LIST_RIWCWP01 Display". The "Enhancement Implementation" field contains "ZUK_EI_WP_LIST_RIWCWP01" and is marked as "Active". The "Enh. Implementation Elements" tab is selected, showing a tree view on the left with "ZUK_IM_WP_LIST_RIWCWP List Processing" expanded, and "Implementing Class" selected. The "Implementing Class" section on the right shows the "Interface" as "IF_EX_BADI_WCM_RIWCWP01_002" and the "Implementing Class" as "ZUK_CL_WP_LIST_RIWCWP01". Below this, a table lists three methods:

Method	Short description
IF_EX_BADI_WCM_RIWCWP01_002~FCODE_DEACTIVATE	Deactivate Function Codes
IF_EX_BADI_WCM_RIWCWP01_002~ALV_TABLES_GET	Set Grouping of Fields and Field Catalog
IF_EX_BADI_WCM_RIWCWP01_002~APPEND_UPDATE	Fill Customer-Specific Fields

Individual class implementation (three interface methods)

6) BAdI Implementation for List Processing Output: Example

The screenshot shows the SAP 'List Editing - Change Orders with Work Permits' interface. A red circle highlights an additional row in the table header, and a red arrow points to it with the text 'Additional row for customer-specific status consolidation'. The table lists various work permits with columns for Order, Applicatn Typ, Short Text, Valid from, Valid to, Status, Approvals, and System status. The 'Status' column is highlighted in blue.

The 'Change Layout: WCM' dialog box is open, showing the layout configuration for the list. The 'Line 1' tab is selected, and the 'Column content' table is visible. The 'Status' field is highlighted in the 'Column content' table, and a red arrow points to it from the text 'Additional row for customer-specific status consolidation'. The 'Hidden fields' table is also visible, showing fields like User Status, Usage, Gültig, Extension, Priority, Recall Time, Unit, OverallCndtn TechSys, Revision Phase, Functional Location, Equipment, and Train.

- Method ALV_TABLES_GET adds customer-specific field to ALV field catalog.
- Method APPEND_UPDATE calculates the field value (not stored on DB).

Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and single WCM Objects (EhP3)
- 3. BAdIs for Additional Data of single WCM Objects (EhP3)
- 4. BAdIs for Screen Enhancements of single WCM Objects (EhP3)
- 5. BAdIs for Menu Enhancements of single WCM Objects (EhP3)
- 6. BAdIs for WCM List Processing Output (EhP5)
- **7. BAdIs for Menu Enhancements of WCM List Processing (EhP5)**
- 8. General Design Principles for WCM BAdIs

7) BAdIs for Menu Enhancements of WCM List Processing

- In each case one BAdI for menu enhancements of **native list processing** output

- 1) of WAPs

- 2) of applications

- 3) of WCDs (both Operational WCD + WCD template)

- 4) of operational lists

- 5) of lineups

- In each case one BAdI for menu enhancements of list selection output **during assignment**

- 1) of orders

- 2) of WAPs

- 3) of applications

- 4) of Operational WCDs

- One BAdI for menu enhancements of WCD list selection output **within clipboard**

- One BAdI for menu enhancements of the combined **work permit/order list** processing output

- One BAdI for menu enhancements of **multilevel list** processing output



7) BAdIs for Menu Enhancements of WCM List Processing

- BAdIs for menu enhancements of WCM list processing are built identically for all WCM objects. They consist of **two basic parts**:
 - The **function codes** provided for the menu enhancement of WCM list processing are integrated in the “Extras” menu on the output list. It is possible to define up to three customer-specific function codes.
 - Furthermore, menu enhancement of a WCM object requires implementing the following **interface method** of the underlying BAdI:
 - FCODE_EXECUTE: Execute a customer-specific function
- Regarding design and methods to be implemented, menu enhancements of WCM list processing is identical to menu enhancements of single WCM objects.



Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and single WCM Objects (EhP3)
- 3. BAdIs for Additional Data of single WCM Objects (EhP3)
- 4. BAdIs for Screen Enhancements of single WCM Objects (EhP3)
- 5. BAdIs for Menu Enhancements of single WCM Objects (EhP3)
- 6. BAdIs for WCM List Processing Output (EhP5)
- 7. BAdIs for Menu Enhancements of WCM List Processing (EhP5)
- **8. General Design Principles for WCM BAdIs**

8) General Design Principles

- In general, the BAdIs provided for customer-specific checks constitute a strengthening as they do not replace the standard checks, but rather complement them.
 - As a matter of principle, a check BAdI is only called after related standard checks have been (successfully!) processed.
- The BAdI import interfaces are kept rather lean, i.e. they do not provide all and every data that might be required during BAdI processing, but only data that is obviously relevant in the respective semantic context.
 - Access to further data within a BAdI implementation can be gained by using the WCM function modules with suffix GLOBAL_DATA_GET.
- Note that beyond its semantic context, each BAdI interface can be considered as a dedicated point-of-time allowing customer-specific control.
 - For example, a BAdI interface for checking if a dialog window can be closed could be used for other purposes instead, e.g. for updating customer-specific data depending on the dialog window settings.


Questions and Answers



Search

Company Product **Services** News Links

Consulting Training **Forums** FAQs Downloads




Deutsch
Contact
Home

International Forums


Board index [View unread posts](#) | [View new posts](#) | [View your posts](#)

PM-WCM is a component of SAP ERP



Restricted Area
Username: Michael Lesk

Change User Profile
Change Password
Logout



Mark forums read				
	Forum	Topics	Posts	Last post
	General Topics	7	19	30 Mar 2010 22:42 Janet Graham, Canada (ID 260) ➔
	WCM-Relevant Orders	8	22	30 Mar 2010 13:48 Harshvardhan Bajad, India (ID 501) ➔
	WCM Objects	20	58	18 Mar 2010 10:29 lucio martino, Italy (ID 526) ➔
	Lockout/Tagout	16	52	22 Feb 2010 01:55 Michael Lesk, Germany (ID 98) ➔
	Mode-Dependent Tagging/Lineup	6	11	25 Mar 2010 06:32 augustine Spivak, United States (ID 256) ➔
	Mobile Processing	2	5	05 Sep 2009 13:03 Christoph Wobbe, Germany (ID 94) ➔
	Business Add-Ins (BADIs)	3	6	25 Feb 2010 14:18 Michael Lesk, Germany (ID 98) ➔

- No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of WCM GmbH.
- The information contained in this publication may be changed by WCM GmbH without prior notice.
- SAP, R/3, mySAP, SAP NetWeaver and other mentioned SAP products and services 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 mentioned product and service names as well as the associated logos are the trademarks of their respective companies.