BAdIs in Work Clearance Management

Release ERP 6.0, EhP 3

Michael Lesk

WCM Info Day, October 2009
Munich, Germany

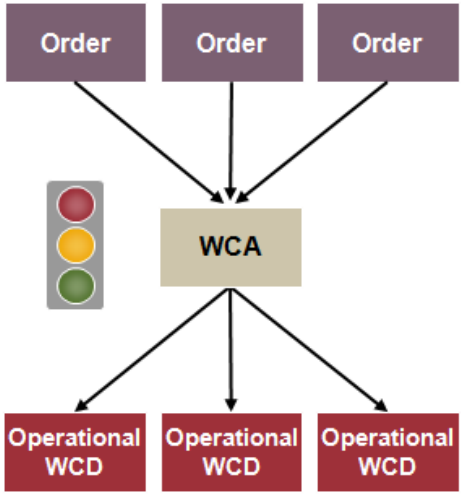
Agenda

- **1. Introduction**
- 2. Process-controlling BAdIs for Order and WCM Objects
- 3. BAdIs for Additional Data of WCM Objects
- 4. BAdIs for Screen Enhancements of WCM Objects
- 5. BAdIs for Menu Enhancements of WCM Objects
- 6. General Design Principles for WCM BAdIs

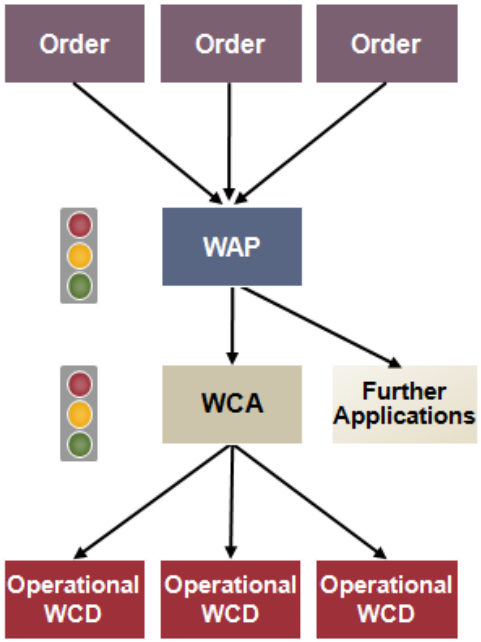
1) Motivation

- Situation before Release ERP 6.0, EhP 3:
  - Not all individual customer requirements are 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, EhP 3:
  - In early 2007, SAP had a WCM info day in Amsterdam, used as platform for the roll-in of customer requirements; in particular, participants were asked to provide SAP with information on their individual functional enhancements.
  - Based on this information, SAP has provided various Business-Add-Ins (BAdIs) with release ERP 6.0, EhP 3, enabling SAP customers to develop individual WCM enhancements without needing to modify the SAP standard.
    - Hence future WCM upgrades can be imported without endangering existing enhancements or needing to adjust them (→ SPAU).

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



The Standard Model

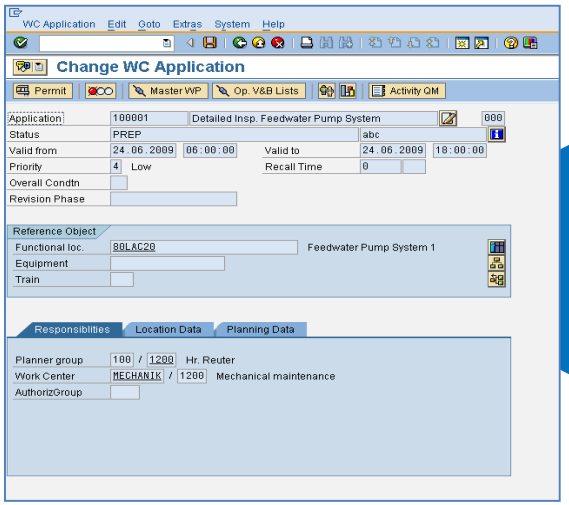


The Enhanced Model

- 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.

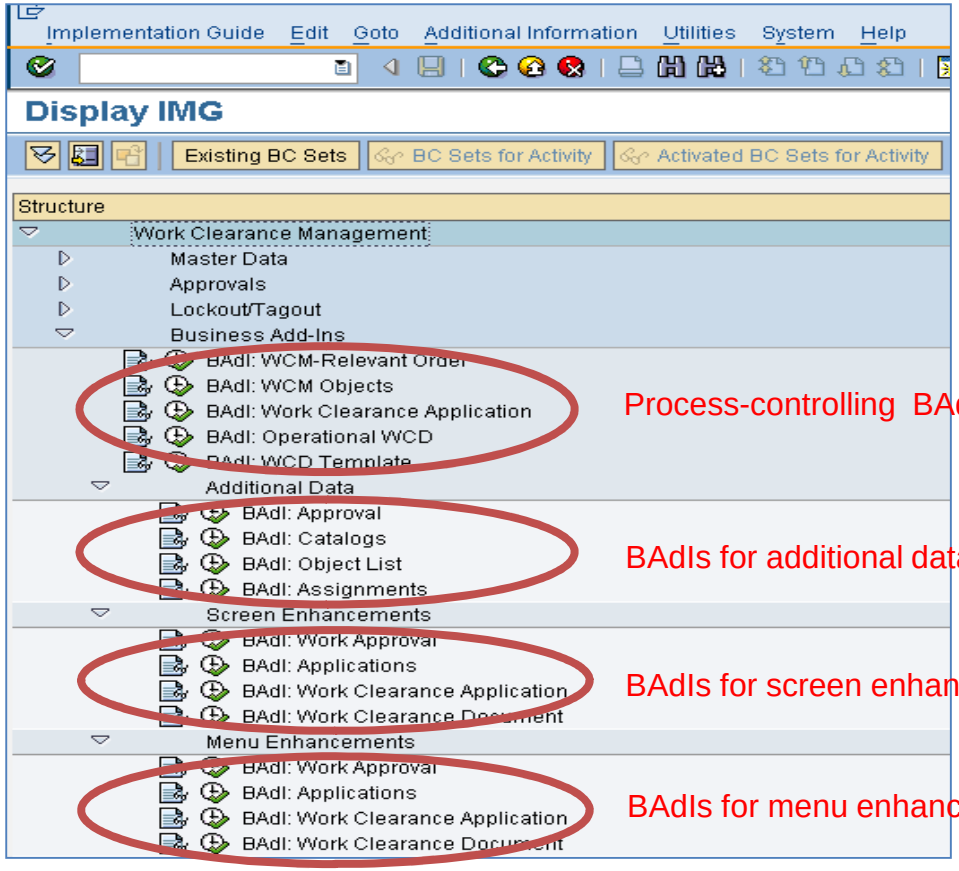
1) ...and continue with a look at the WCM Objects



- 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

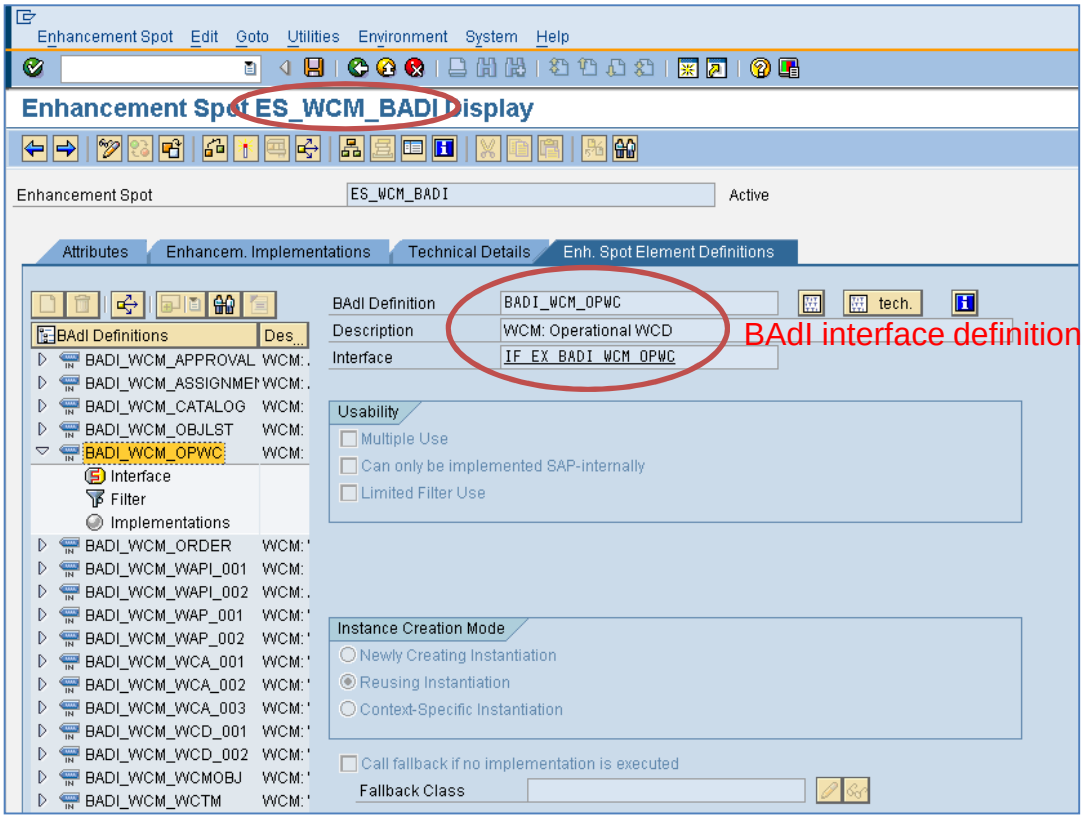
- For processing WCM Objects like e.g. the WCA, customers asked for the option to
  - enhance the screens by individual fields → SAP provides screen BAdIs
  - enhance the menus by individual functions → SAP provides menu BAdIs
  - control processing of additional data → SAP provides BAdIs for additional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

1) Let‘s have a first look at the BAdIs available in IMG…



- Each WCM BAdI interface (definition + implementation) can be accessed via IMG.
  - Note that access via IMG requires activation of the EAM business function LOG_EAM_CI_2 in the Switch Framework.

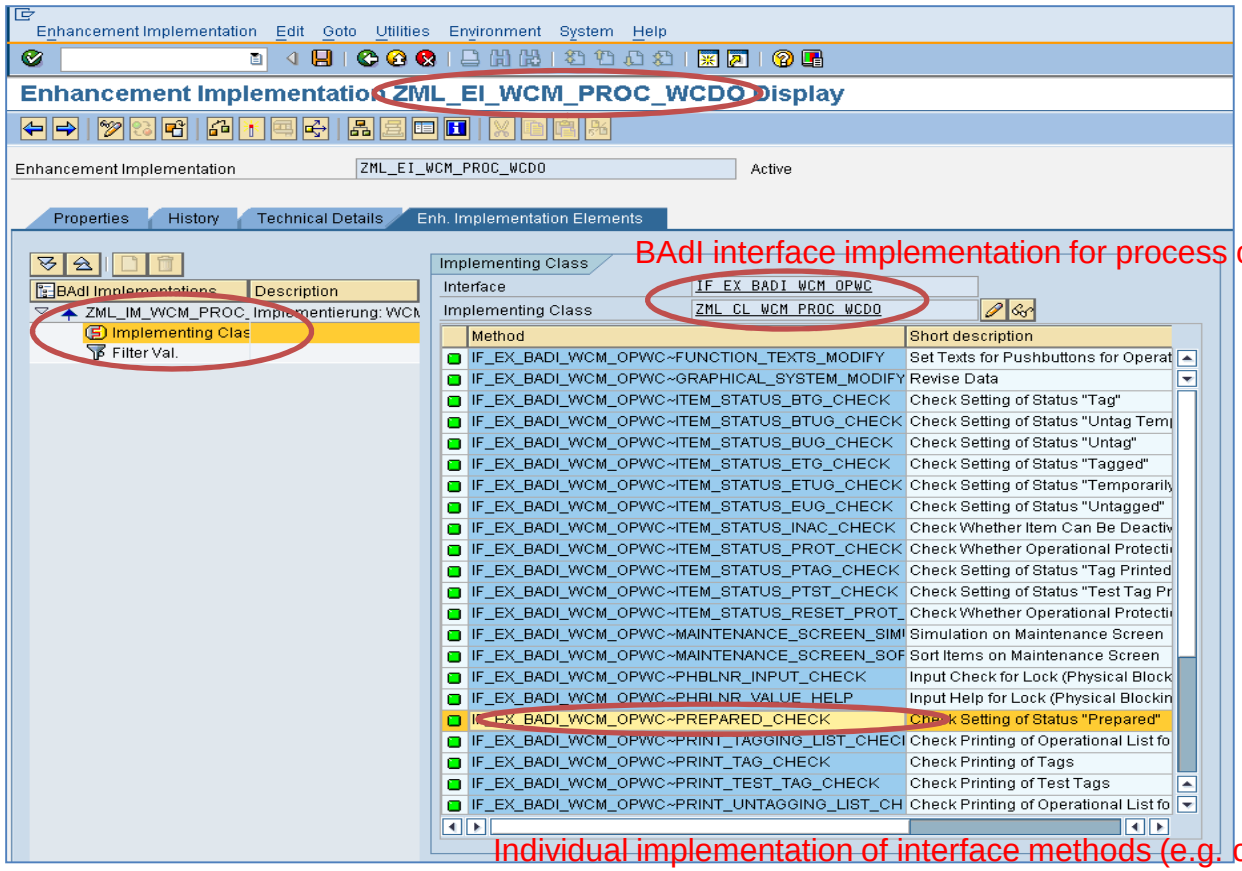
1) ...and continue with a look at the BAdI Builder...



BAdI interface definition for process control of Operational WCDs

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

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



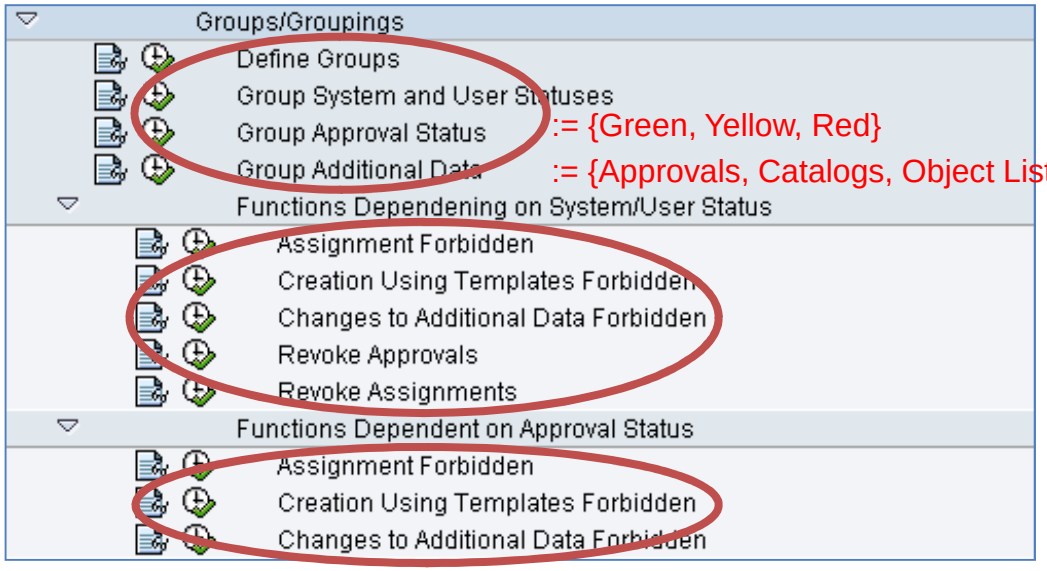
BAdI interface implementation for process control of Operational WCDs

Individual implementation of interface methods (e.g. check of WCD preparation)

- A WCM enhancement implementation may contain several implementing classes.
- 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, EhP 4)
  - Enabled via Customizing, no additional implementation required.
  - IMG:



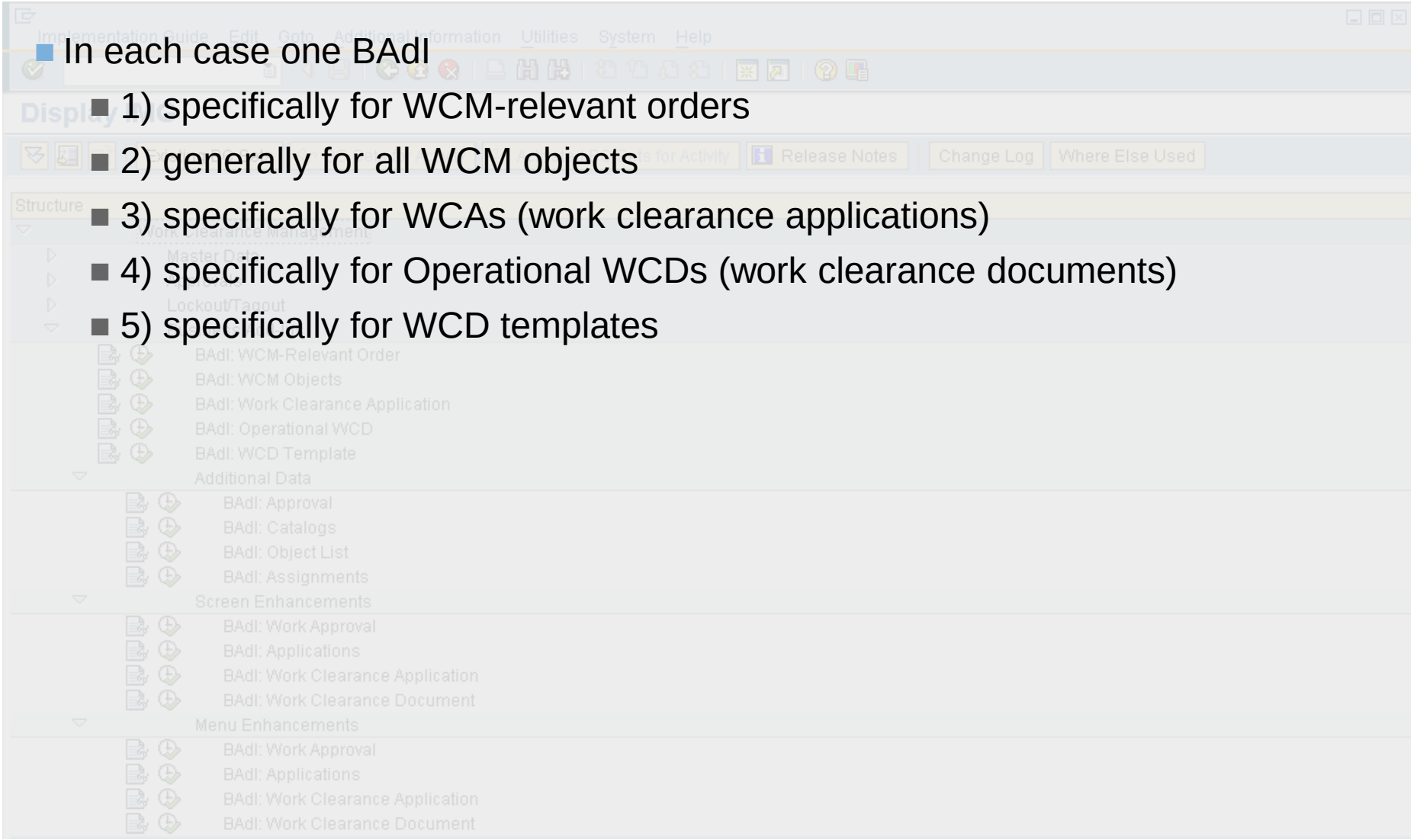
  - Example: Revoke final safety approval in WCA when test cycle is triggered.

Agenda

- 1. Introduction
- **2. Process-controlling BAdIs for Order and WCM Objects**
- 3. BAdIs for Additional Data of WCM Objects
- 4. BAdIs for Screen Enhancements of WCM Objects
- 5. BAdIs for Menu Enhancements of WCM Objects
- 6. General Design Principles for WCM BAdIs

2) Process-controlling BAdIs for Order and WCM Object

- 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 valuated as requested (“Yes”, “No”)
  - Check if the valuation dialog can be closed as requested (by the green check)

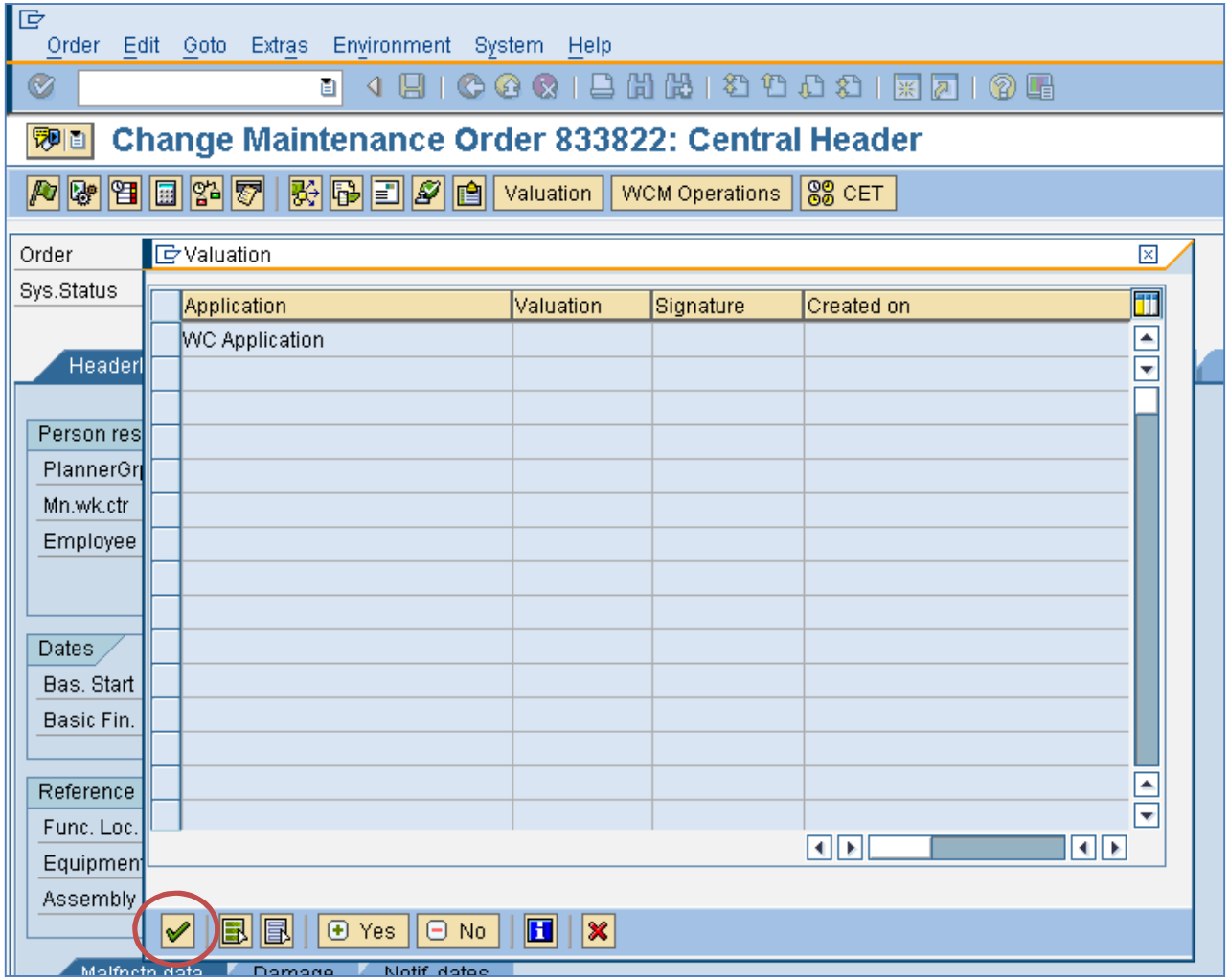
- Methods for work release/completion in a WCM-relevant order:
  - Check if “Release for Execution” can be issued as requested
  - Check if “Work Completed” can be confirmed as requested

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)

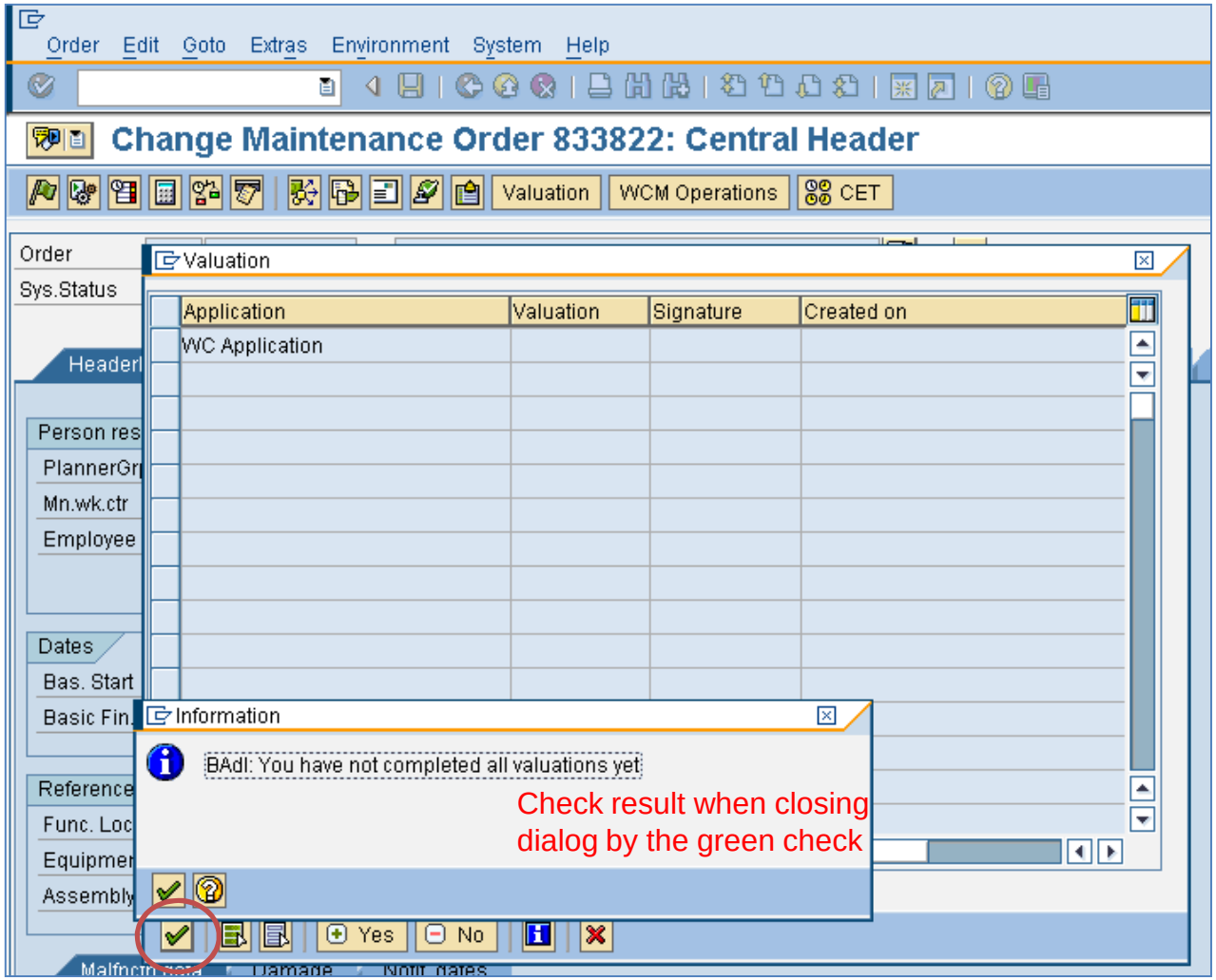
- Methods for work release/completion of WCM-relevant operations:
  - Check if “Release for Execution” can be issued as requested
  - Check if “Work Completed” can be confirmed as requested

2.1) Example 1
Check if the Valuation Dialog can be closed as requested (1/2)

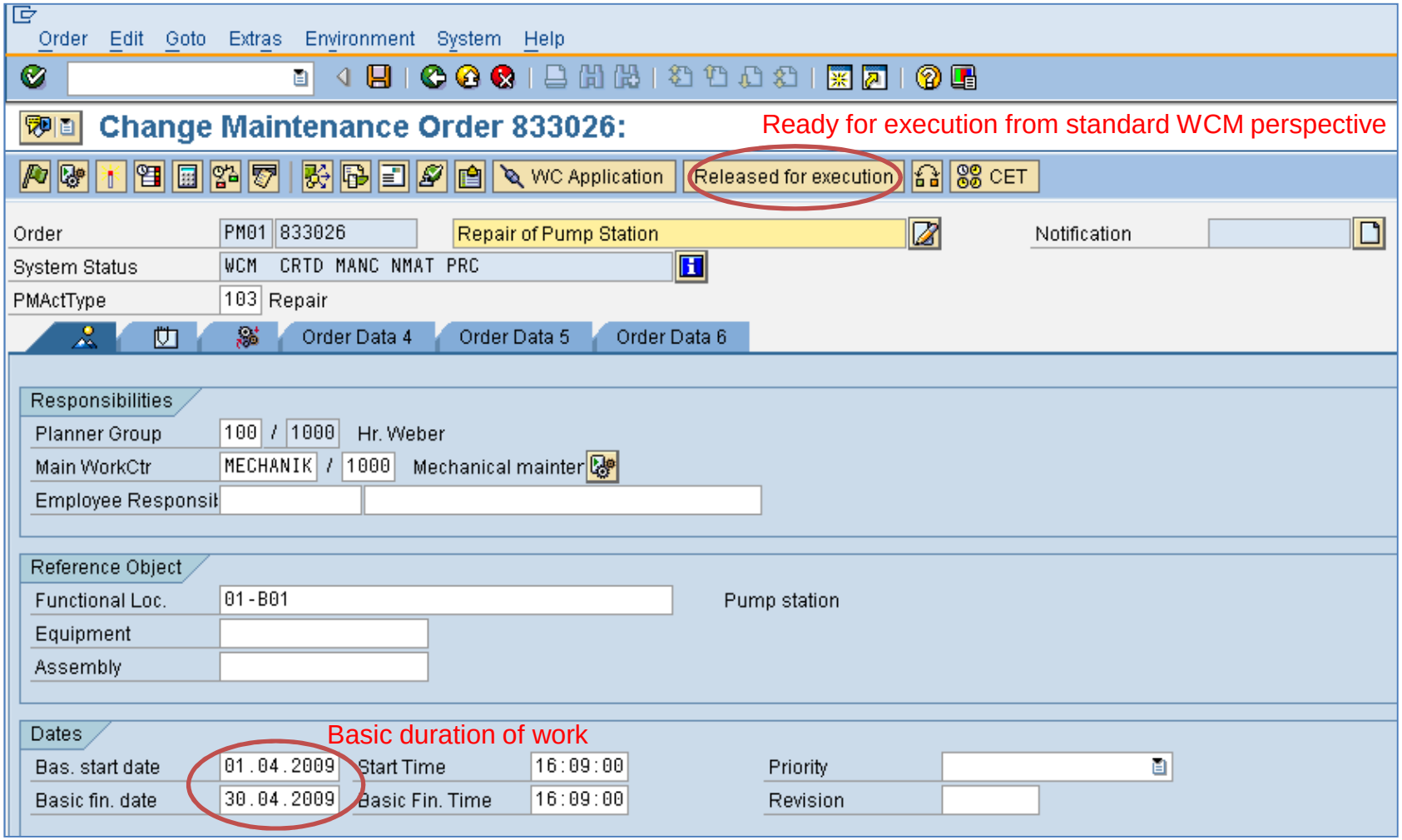


Green check for leaving valuation dialog

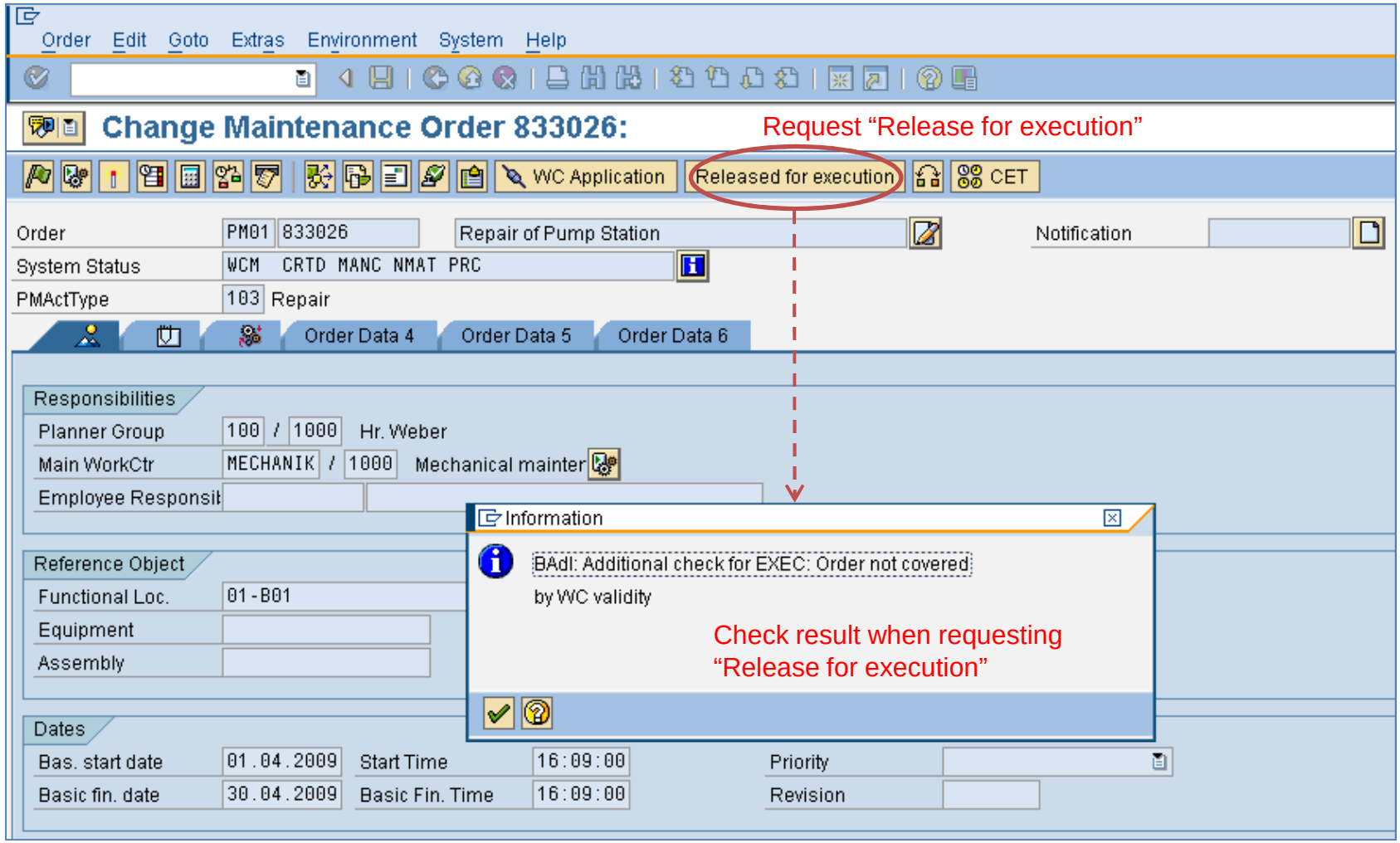
2.1) Example 1
Check if the Valuation Dialog can be closed as requested (2/2)



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



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



2.2) Process-controlling BAdI for all WCM Objects General Functions

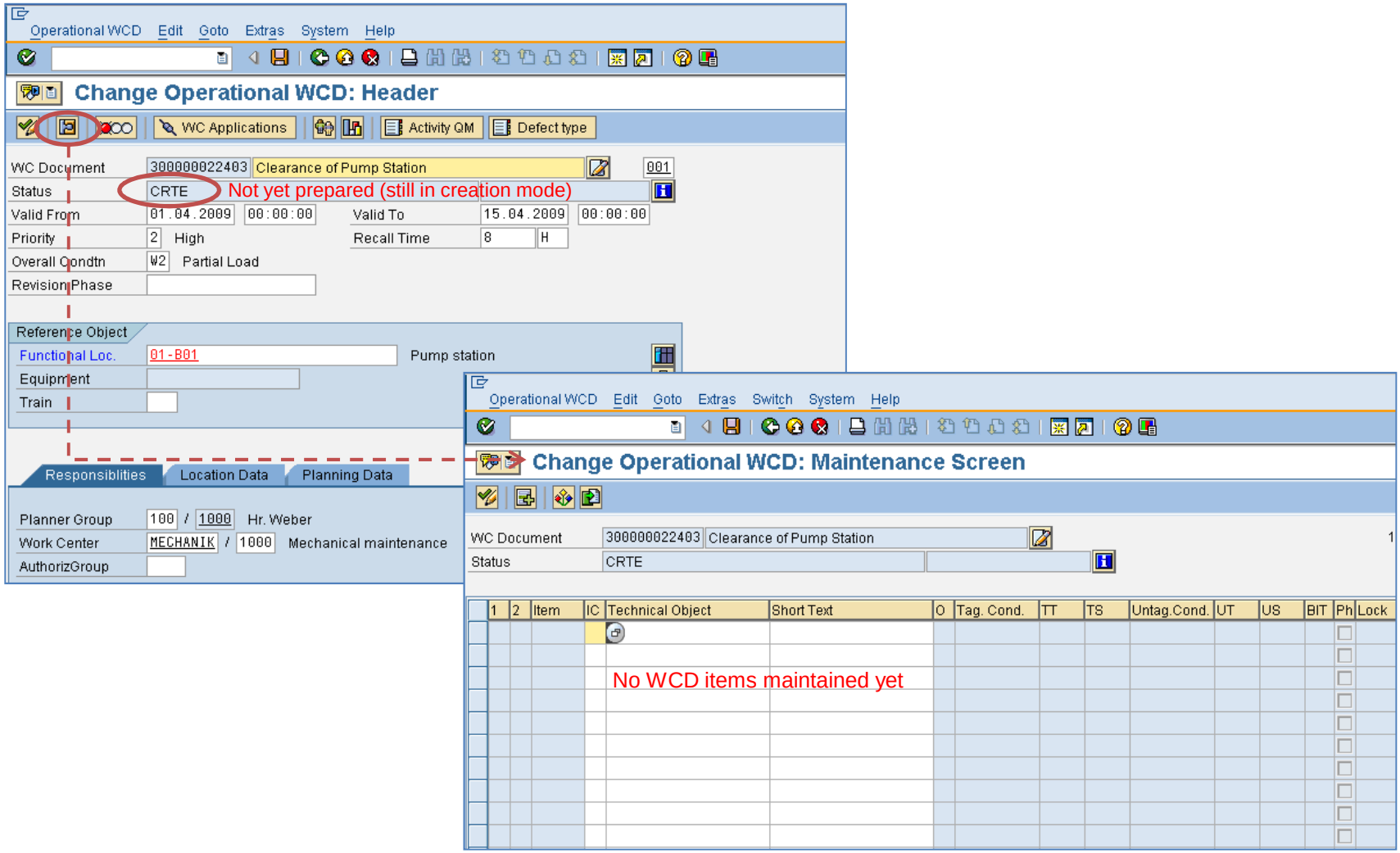
- General BAdI methods for WCM objects:
  - Change color or short text (e.g. in multi-level list)
  - Set user field for standard field selection
  - Set change indicator for WCM object (e.g. for confirmation prompt upon exit)
  - Check WCM object in case of requested save

- 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

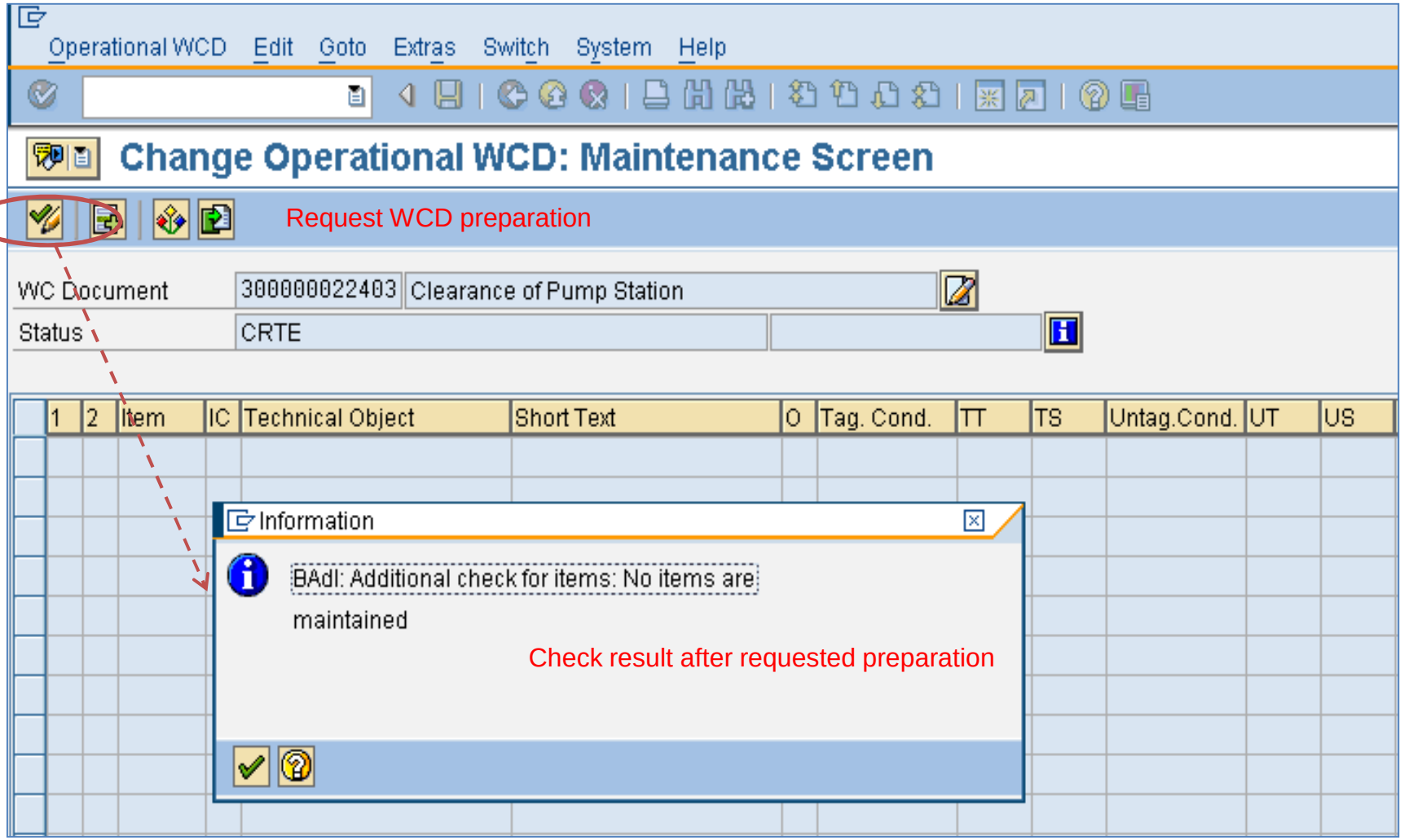
2.2) Process-controlling BAdI for all WCM Objects Status Changes

- 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

2.2) Example
Check if Operational WCD can be prepared as requested (1/2)



2.2) Example
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 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

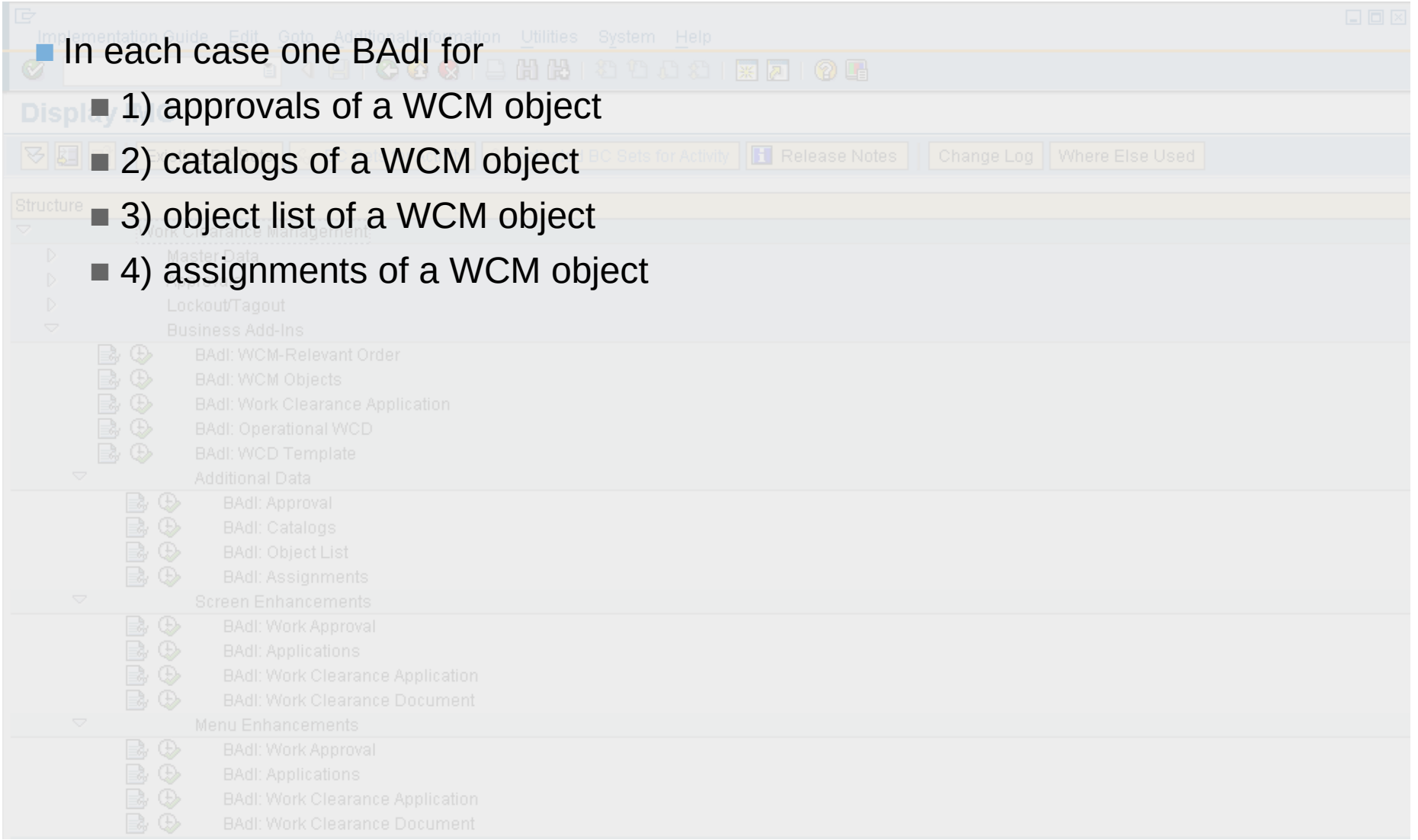
- Methods for WCD templates:
  - Default sort sequence of the items of a WCD template on the maintenance screen
  - Extend the standard simulation on the maintenance screen by individual checks
  - Check if the status of a WCD item can be set to “Inactive” as requested

Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and WCM Objects
- **3. BAdIs for Additional Data of WCM Objects**
- 4. BAdIs for Screen Enhancements of WCM Objects
- 5. BAdIs for Menu Enhancements of WCM Objects
- 6. General Design Principles for WCM BAdIs

3) BAdIs for additional data of WCM Objects

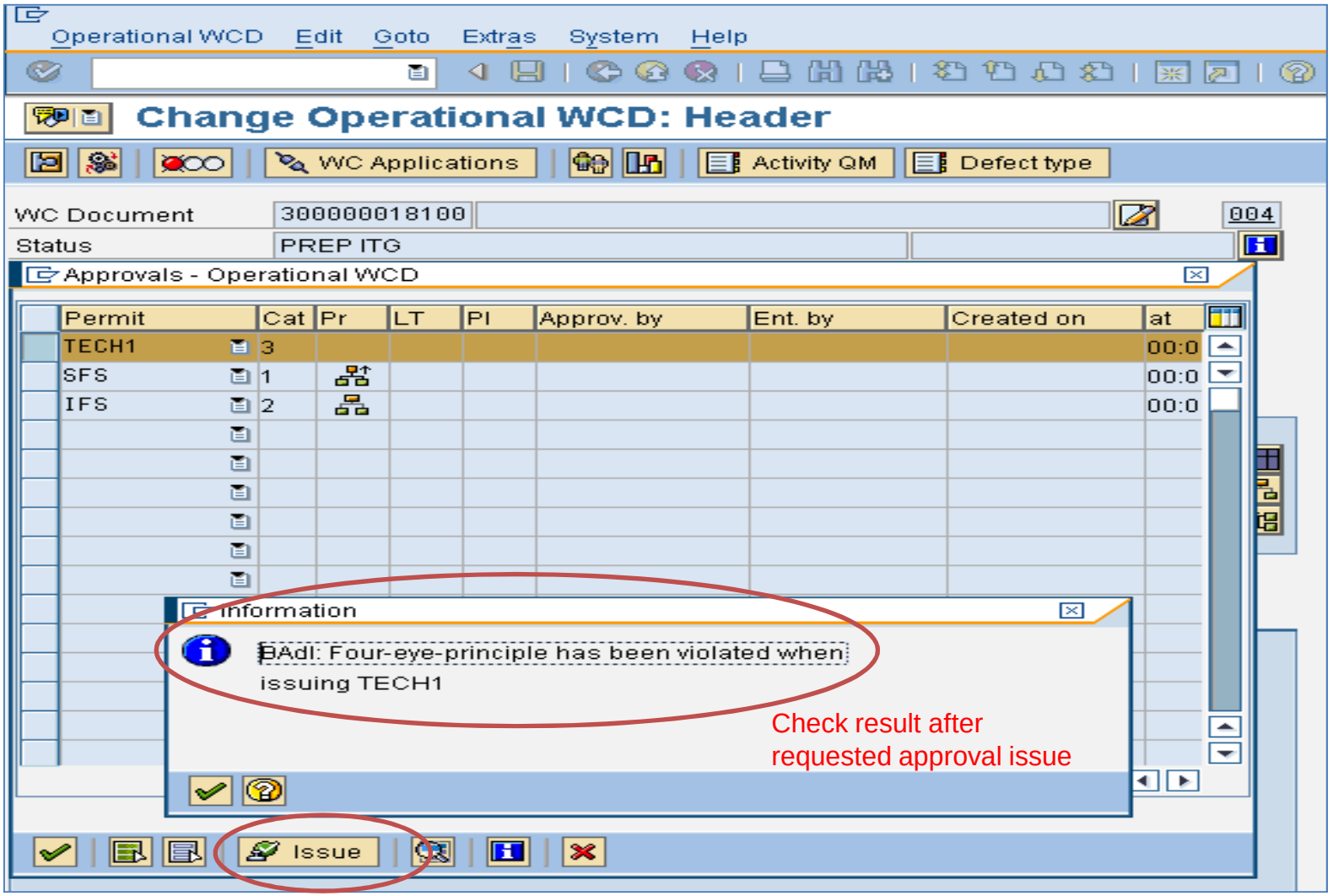
- In each case one BAdI for
  - 1) approvals of a WCM object
  - 2) catalogs of a WCM object
  - 3) object list of a WCM object
  - 4) assignments of a WCM object



3.1) BAdI for approvals of WCM Objects

- 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

3.1) Example
Check if Approval can be issued 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 valuated 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

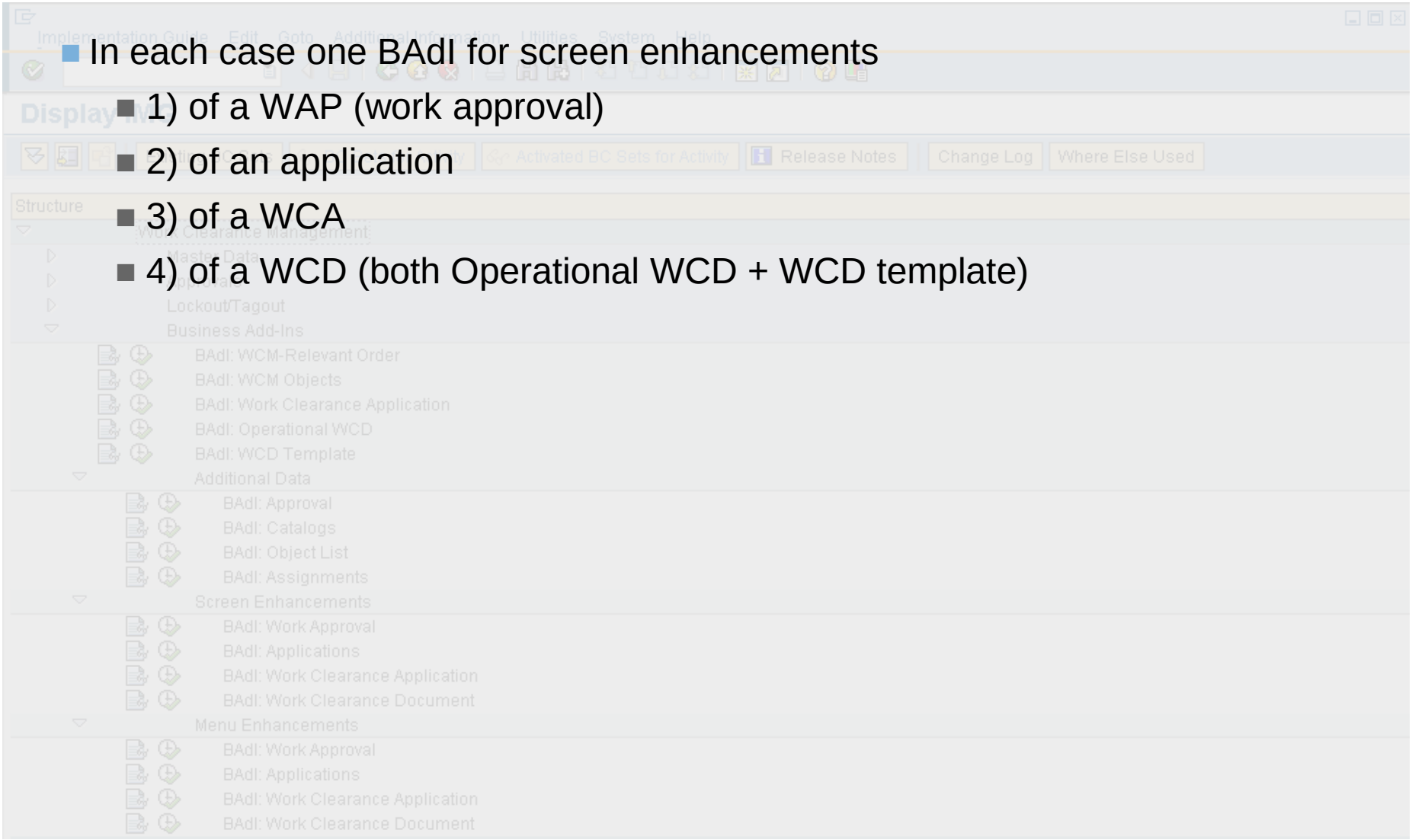
- The BAdI provides the following implementation options:
  - Deactivate functions for maintaining assignments
  - Check if a maintained assignment can be removed as requested
  - Check if the dialog window for maintaining superior and subordinate assignments can be closed as requested
  - Restrict the 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 WCM Objects
- 3. BAdIs for Additional Data of WCM Objects
- **4. BAdIs for Screen Enhancements of WCM Objects**
- 5. BAdIs for Menu Enhancements of WCM Objects
- 6. General Design Principles for WCM BAdIs

4) BAdIs for Screen Enhancements of a WCM Object

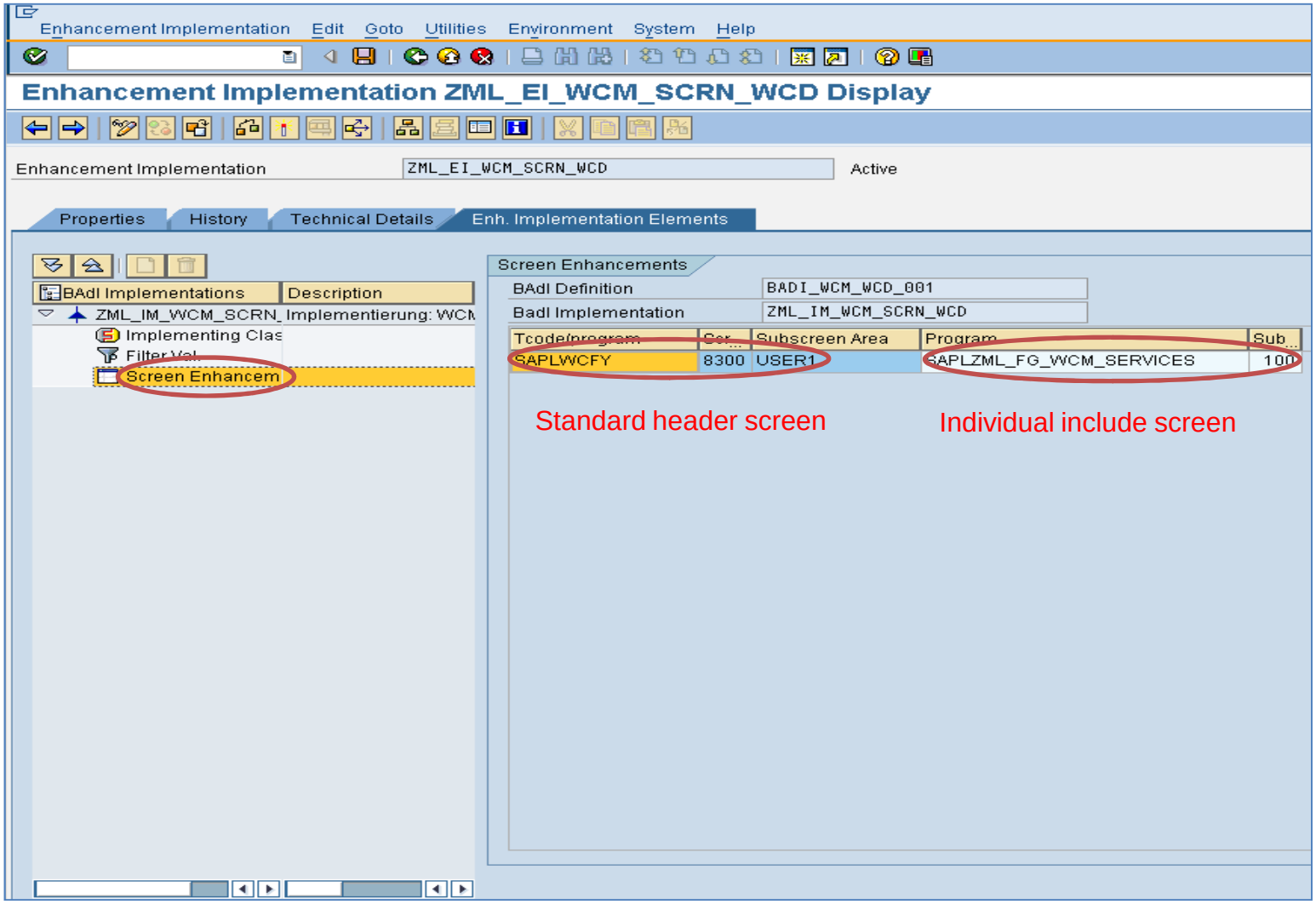
- In each case one BAdI for screen enhancements
  - 1) of a WAP (work approval)
  - 2) of an application
  - 3) of a WCA
  - 4) of a WCD (both Operational WCD + WCD template)



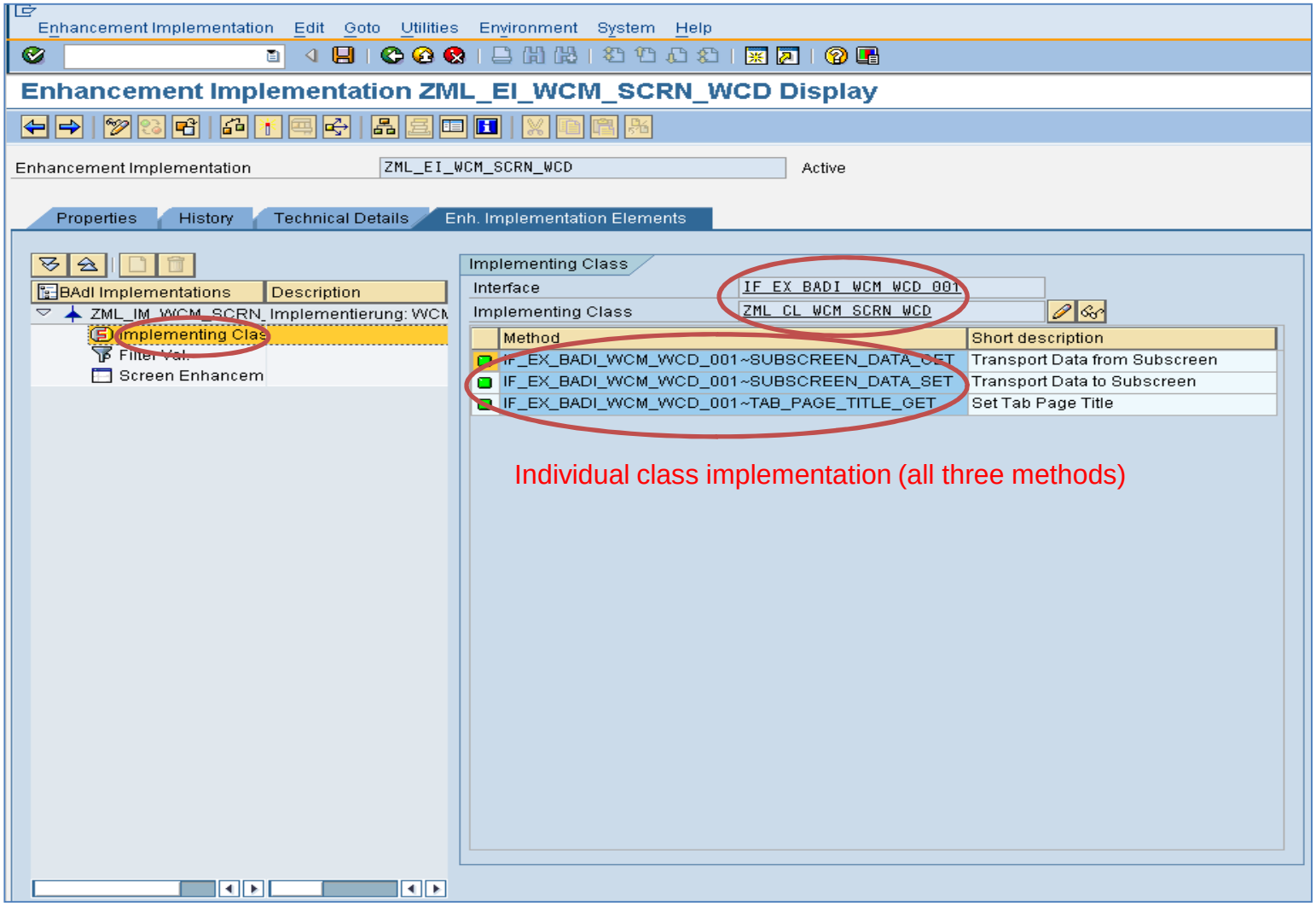
4) BAdIs for Screen Enhancements per WCM Object

- 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 is integrated as fourth 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.
    - Note that as of EhP 5, flexible arrangement of all WCM object header subscreens (including the customer-specific one) will be supported.
  - 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 fourth tab on the header screen

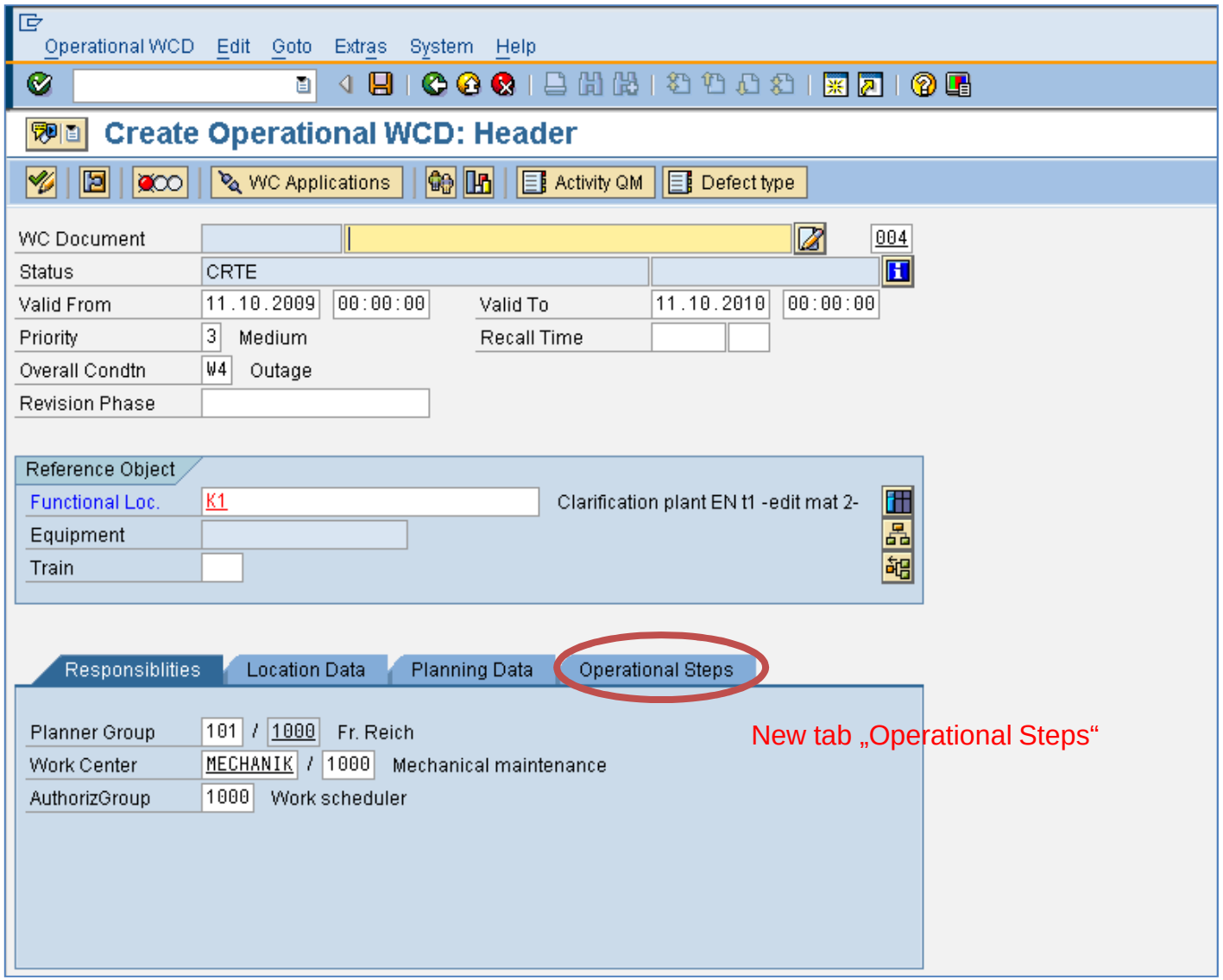
4) Example
Add additional Tab Page to Screen of Operational WCD (1/3)



4) Example
Add additional Tab Page to Screen of Operational WCD (2/3)



4) Example
Add additional Tab Page to Screen of Operational WCD (3/3)

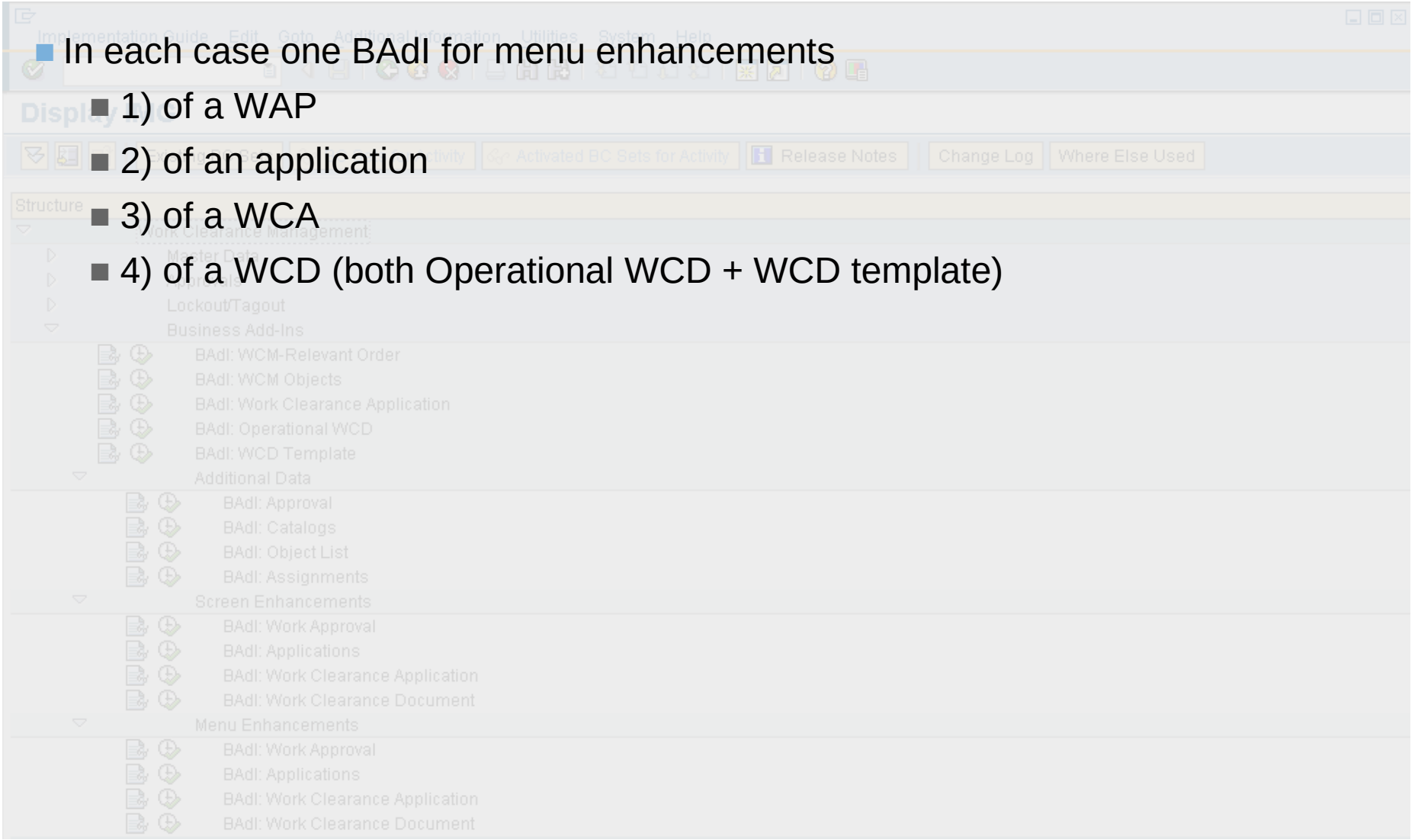


Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and WCM Objects
- 3. BAdIs for Additional Data of WCM Objects
- 4. BAdIs for Screen Enhancements of WCM Objects
- **5. BAdIs for Menu Enhancements of WCM Objects**
- 6. General Design Principles for WCM BAdIs

5) BAdIs for Menu Enhancements of a WCM Object

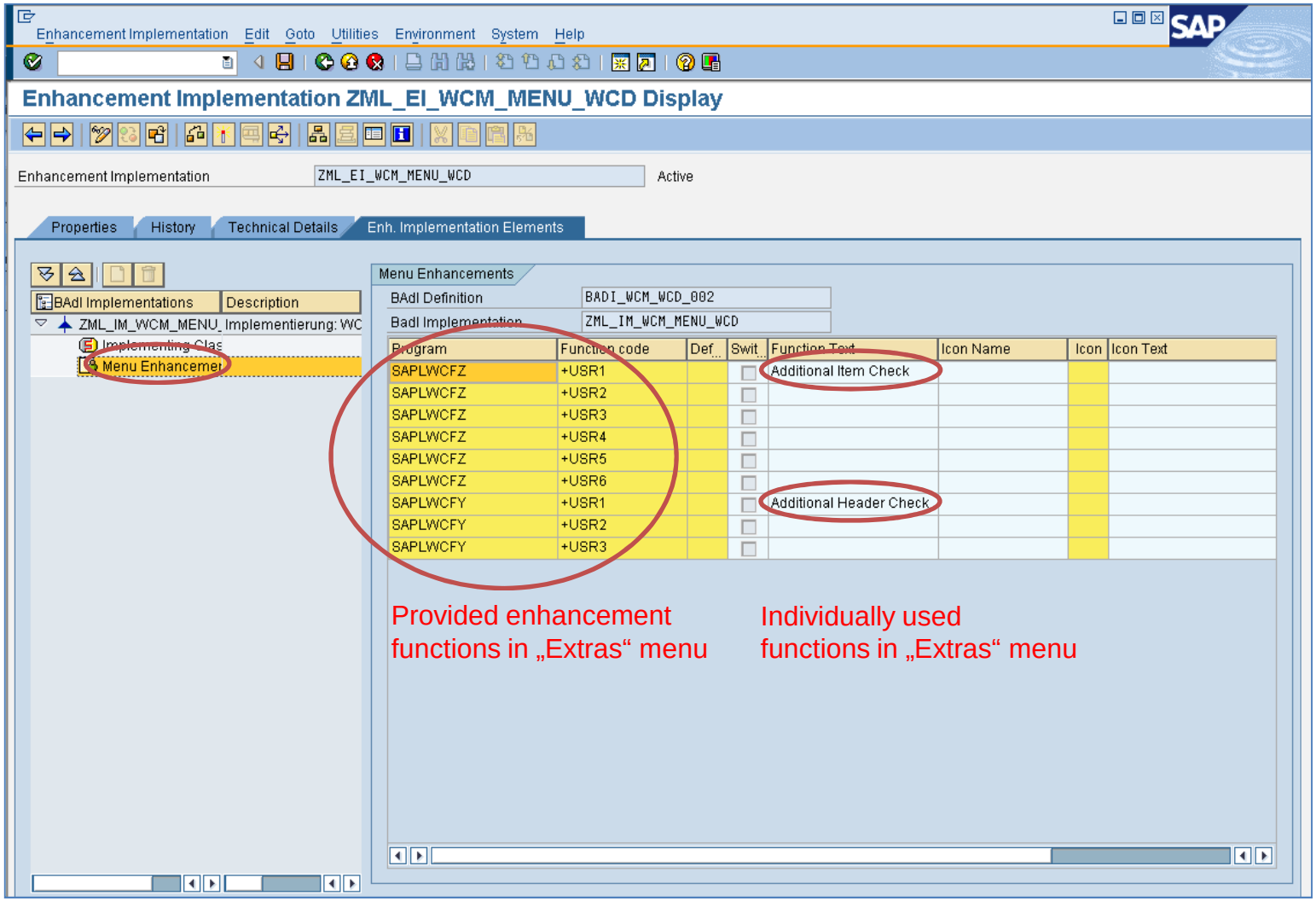
- In each case one BAdI for menu enhancements
  - 1) of a WAP
  - 2) of an application
  - 3) of a WCA
  - 4) of a WCD (both Operational WCD + WCD template)



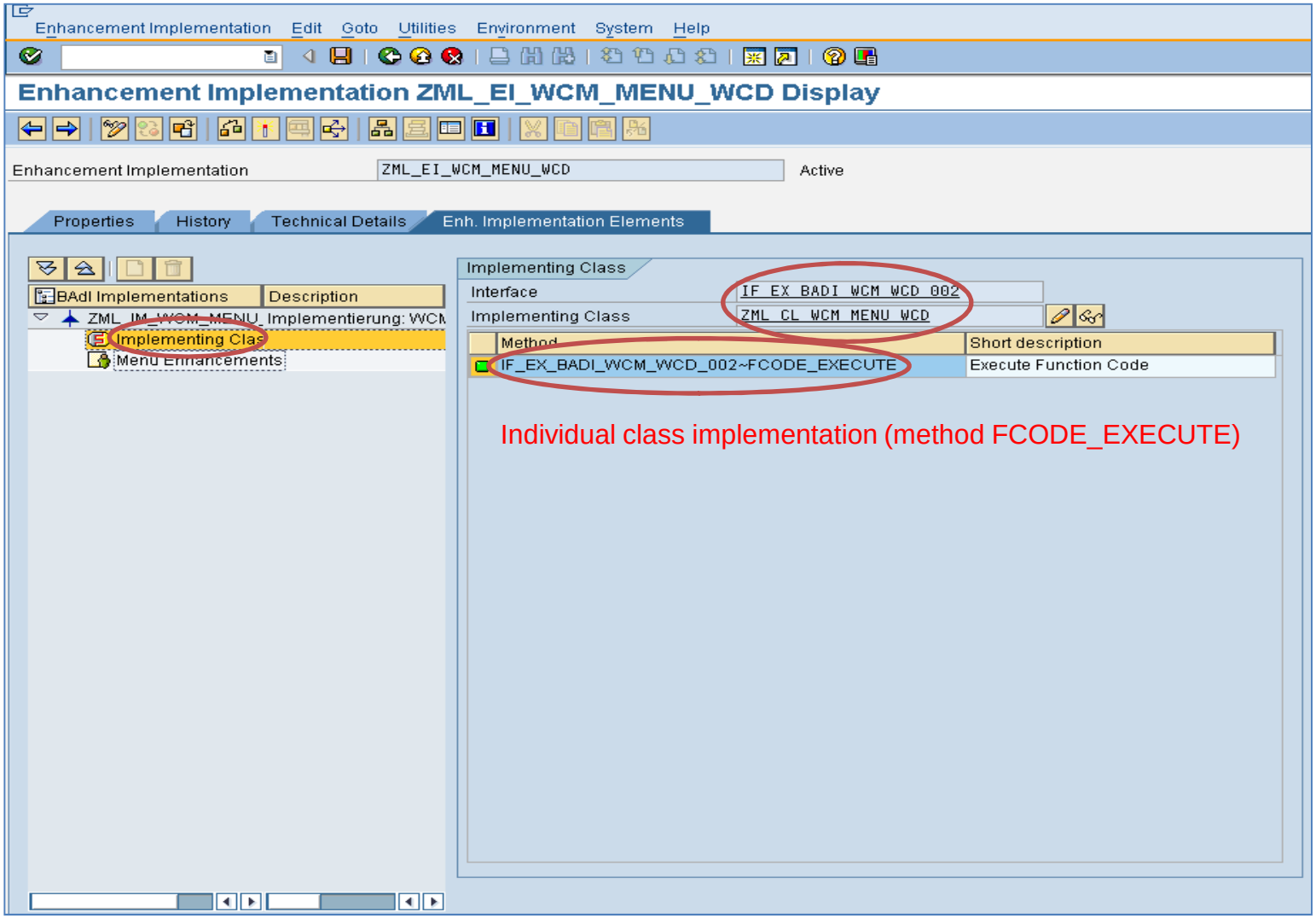
5) BAdIs for Menu Enhancements per WCM Object

- Menu BAdIs are built identically for all WCM objects. They consist of two basic parts:
  - The functions 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

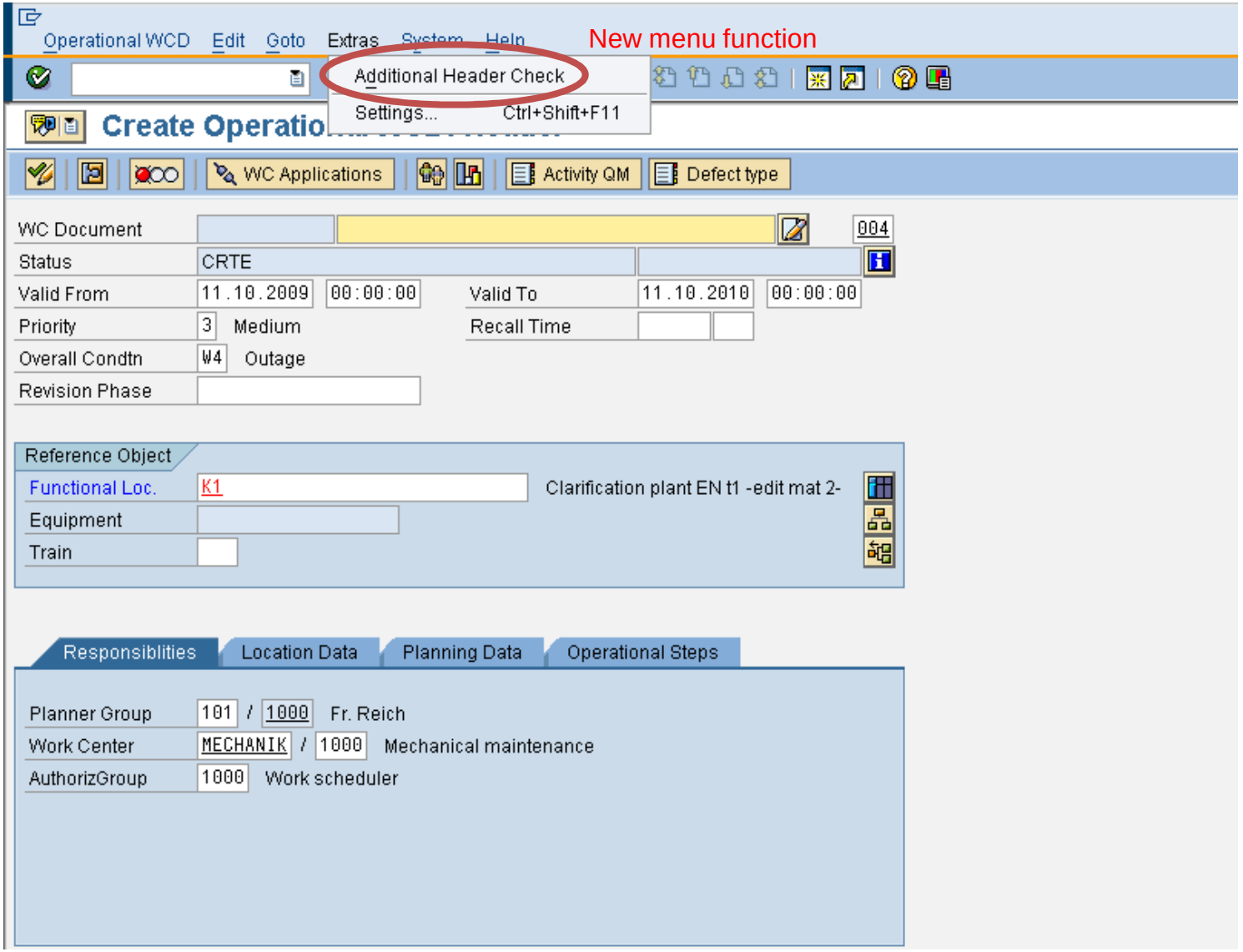
5) Example
Add additional Function to Menu of Operational WCD (1/3)



5) Example
Add additional Function to Menu of Operational WCD (2/3)



5) Example
Add additional Function to Menu of Operational WCD (3/3)



Agenda

- 1. Introduction
- 2. Process-controlling BAdIs for Order and WCM Objects
- 3. BAdIs for Additional Data of WCM Objects
- 4. BAdIs for Screen Enhancements of WCM Objects
- 5. BAdIs for Menu Enhancements of WCM Objects
- **6. General Design Principles for WCM BAdIs**

6) 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 specific 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

Copyright 2009 WCM GmbH