

## AGIMENDO.edit

---

### Handbuch und Anwenderdokumentation

Produktversion	v2.15.0
Handbuchversion	v1.7.0
Status	Freigegeben
Datum	02.11.2016
Verantwortlich	Felix Weyde (felix.weyde@ibsolution.de)

## 1. Inhaltsverzeichnis

<b>1. Inhaltsverzeichnis .....</b>	<b>2</b>
<b>2. Einführung.....</b>	<b>5</b>
2.1 Funktionsweise.....	5
2.2 Zusammenhang und Komponenten.....	6
<b>3. Zuordnung von Benutzer bzw. Rollen zu einem View .....</b>	<b>6</b>
3.1 Neue View anlegen.....	6
3.2 Benutzer zu einem View zuordnen .....	7
<b>4. Wizard-basierte Verwaltung von View Inhalten .....</b>	<b>8</b>
4.1 Auswahl eines Views .....	8
4.2 Auswahl des zu bearbeitenden Merkmals .....	9
4.3 Viewpflege.....	10
4.3.1 Änderungsverhalten.....	12
4.3.2 Tabellendarstellung.....	13
4.3.3 Übersetzungsunterstützung.....	14
4.3.4 Spaltenerweiterungen.....	15
4.3.5 Technische Einstellungen. ....	17
4.3.6 Konfiguration des für die Pflege zugeordneten Attributs .....	18
<b>5. Stammdatenpflege .....</b>	<b>19</b>
5.1 Aufruf der Stammdatenpflege .....	19
5.2 Parametrisierter Aufruf der Stammdatenpflege.....	21
<b>6. AGIMENDO.edit User Exits .....</b>	<b>23</b>
6.1 Definition und Ablage von User Exits .....	23
6.2 AGIMENDO.edit User Exits pflegen.....	23
6.3 Übersicht der AGIMENDO.edit User Exits .....	26
6.4 AGIMENDO.edit Ausführungsreihenfolge User Exits .....	27
6.4.1 User Exits bei der Auswahl des zu bearbeitenden Merkmals .....	27
6.4.2 User Exits beim Anlegen von Datensätzen.....	28
6.4.3 User Exits beim Speichern von Datensätzen.....	29
6.4.4 User Exits beim Zwischenspeichern von Datensätzen .....	30
6.4.5 User Exits beim Löschen von Datensätzen .....	30
<b>7. Berechtigungen.....</b>	<b>31</b>
<b>8. Customizing .....</b>	<b>31</b>
<b>9. Transportwesen.....</b>	<b>32</b>
9.1 Überblick .....	32

9.1.1	Trennung der Systeme .....	32
9.2	Transportieren über den Wizard .....	33
9.2.1	Transport eines vollständigen Views.....	33
9.2.2	Transport eines einzelnen InfoObjekts in einem View.....	33
9.3	Transportmanager .....	35
9.3.1	Ausgehende Transportitems .....	36
9.3.2	Eingehende Transportitems .....	40
9.3.3	Transportwesen als Backup.....	40
9.4	Altes Transportvorgehen / vollständiger Transport.....	41
<b>10.</b>	<b>Anhang .....</b>	<b>43</b>
10.1	HowTo-Sammlung zur UserExit-Implementierung.....	43
10.1.1	Anlage mit Vorlage .....	43
10.1.2	Ausschluss von Merkmalen von der Gültigkeitsprüfung.....	43
10.1.3	Ändern der Sortierspalten einer Tabelle .....	46
10.2	Referenz der UserExit.....	47
10.2.1	AFTER_ATTR_CHG (Methode: AFTER_ATTR_CHG) .....	47
10.2.2	AFTER_DELETE (Methode: AFTER_DELETE) .....	47
10.2.3	AFTER_SAVE (Methode: AFTER_SAVE).....	47
10.2.4	BEFORE_ATTR_CHG (Methode: BEFORE_ATTR_CHG).....	47
10.2.5	BEFORE_DELETE (Methode: BEFORE_DELETE) .....	48
10.2.6	BI_AUTHORISATION (Methode: CHECK_BI_AUTHORITY) .....	48
10.2.7	BI_AUTH_CREATE (Methode: CHECK_BI_AUTHORITY_CREATE) .....	49
10.2.8	CHANGE_ALV (Methode: CHANGE_ALV) .....	49
10.2.9	PFCG_AUTHORISATION (Methode: CHECK_PFCG_AUTHORITY).....	49
10.2.10	CUSTOM_AUTHORISATION (Methode: CHECK_CUSTOM_AUTHORITY).....	50
10.2.11	CUSTOM_AUTH_CREATE (Methode: CHECK_CUSTOM_AUTH_CREATE).....	50
10.2.12	CUSTOM_POPUP_FIELD (Methode: CUSTOM_POPUP_FIELD) .....	51
10.2.13	CUSTOM_VALIDATION (Methode: CHECK_VALIDATION) .....	51
10.2.14	EXECUTE_BTN_ACTION (Methode: EXECUTE_BTN_ACTION) .....	52
10.2.15	EXEC_BTN_ACTION_CLOS (Methode: EXEC_BTN_ACTION_CLOSED) .....	53
10.2.16	CUSTOM_VALUE_HELP (Methode: GET_CUSTOM_VALUE_HELP).....	53
10.2.17	FILTER_LANGU (Methode: FILTER_LANGU) .....	54
10.2.18	GET_VALUE (Methode: GET_VALUE) .....	54
10.2.19	GET_KEYS_F_SELECTION (Methode: GET_KEYS_F_SELECTION) .....	55

10.2.20	GET_NEW_LINES_COUNT (Methode: GET_NEW_LINES_COUNT) .....	55
10.2.21	GET_SORT_FULL_TABLE (Methode: GET_SORT_FULL_TABLE) .....	55
10.2.22	PREPAPRE_PREVIEW_TABLE (Methode: PREPARE_PREVIEW_TABLE) .....	56
10.2.23	PREPAPRE_UPLOAD_TABLE (Methode: PREPARE_UPLOAD_TABLE) .....	56
10.2.24	READ_ONLY_MODE (Methode: READ_ONLY_MODE) .....	57
10.2.25	SKIP_ATTRIB_CHANGE (Methode: SKIP_ATTRIB_CHANGE) .....	57
10.2.26	SHOW_BUTTONS (Methode: GET_UE_BUTTONS_FOR_LOCATION) .....	57
10.2.27	TEXT_CHANGED (Methode: TEXT_CHANGED) .....	58
10.3	API Methoden Beschreibung .....	59
10.3.1	GET_CHANGES .....	59
10.3.2	GET_READ_ONLY .....	59
10.3.3	GET_SELECTED_ENTRIES .....	59
10.3.4	GET_SELECTED_ENTRY .....	59
10.3.5	GET_URL_PARAMS .....	60
10.3.6	RAISE_EXTERNAL_BUTTON_EVENT .....	60
10.3.7	READ_TABLE_ROW .....	60
10.3.8	SET_ENTRY .....	60
10.3.9	SET_LEAD_SELECTION .....	60
10.4	Änderungshistorie .....	60
10.5	Abbildungsverzeichnis .....	61

## 2. Einführung

AGIMENDO.edit ist die Lösung für Mitarbeiter im Unternehmen, die verantwortlich dafür sind, dass die Stammdaten im SAP Business Warehouse (BW) korrekt sind. Was bislang nur in Abhängigkeit zur IT-Abteilung möglich war - nämlich das Pflegen der Stammdaten zum Zweck eines aussagekräftigen Reportings oder einer vollständigen Planung - kann nun selbständig vom Fachbereich getätigt werden. Dabei geht es in erster Linie um die Neuanlage von Daten, um Daten, die nicht kontinuierlich durch das Quellsystem aktualisiert werden oder um die Ergänzung von Daten mit Attributen, die nur für das SAP BW relevant sind. Der SAP Standard bedient diese Herausforderungen nur unzureichend, u.a. aufgrund eines fehlenden Berechtigungskonzeptes. AGIMENDO.edit stellt ein eben solches Konzept, sowie die relevantesten Funktionen zur Stammdatenpflege, bspw. Anlegen, Ändern und Löschen bereit.

### Aufbau dieses Dokuments

Für ein Verständnis von AGIMENDO.edit wird im Folgenden zunächst die grundlegende Funktionsweise erläutert. Im nächsten Kapitel wird auf die erste Komponente, das Berechtigungskonzept von AGIMENDO.edit eingegangen. Im vierten Teil werden die verschiedenen Konfigurationsmöglichkeiten beschrieben und erklärt, wie diese mit Hilfe von AGIMENDO.edit abgebildet werden können. In Kapitel 5 wird dargestellt, wie die Pflegeanwendung funktioniert und in Kapitel 6 werden die verschiedenen User-Exits vorgestellt. Zum Schluss werden noch grundlegende Erläuterungen zu den Berechtigungen, sowie zu den Tabellen und dem Transportwesen gemacht.

### Voraussetzungen und Zielgruppe

Dieses Dokument richtet sich an Mitarbeiter und Verantwortliche im Unternehmen, die aktiv mit dem SAP BW arbeiten und den Anspruch haben, Stammdaten anlegen bzw. sie jederzeit eigenständig anpassen zu können. Und dies ganz ohne die Unterstützung der IT-Abteilung.

Für das vollständige Verständnis des Kapitels 6 und die Anwendbarkeit der dort beschriebenen Szenarien und Techniken wird Wissen über ABAP und ABAP-OO benötigt.

## 2.1 Funktionsweise

Ganz allgemein betrachtet ist AGIMENDO.edit eine vereinfachte Möglichkeit, Stammdaten des SAP BW zu pflegen. Dazu gibt es verschiedene Komponenten, die eine leichtere Wartung von Stammdaten ermöglichen. Zum einen gibt es eine Konfiguration von Oberflächen, zum anderen die Berechtigung einzelner Personen für diese Oberflächen. Außerdem findet die Pflege in einer deutlich vereinfachten Maske statt. Somit können auch die Personen aus einem Fachbereich Daten pflegen, die auf technischer Ebene weniger Erfahrung haben.

Die Grundidee ist die Erstellung einer Oberfläche, der verschiedene Benutzer zugewiesen sind. Diese Pflegeoberfläche wird in AGIMENDO.edit als „View“ bezeichnet und wird in Kapitel 3 mit all ihren Konfigurationsmöglichkeiten beschrieben. Die zugeordneten Benutzer sind dazu berechtigt, Stammdaten innerhalb dieser Oberfläche zu pflegen. Welche Stammdaten zur Verfügung stehen, wird über die Zuordnung von Infoobjekten zu dieser Oberfläche geregelt (z.B. Infoobjekt OCUSTOMER). Für diese Infoobjekte kann eine Teilmenge von ihren Attributen ausgewählt und später gepflegt werden (z.B. Stammdaten STADT und Text TXTMD). Außerdem kann die Oberfläche individuell eingerichtet werden, damit eine übersichtliche Maske im Pflegemodus zur Verfügung

steht. Damit ist AGIMENDO.edit in vielerlei Hinsicht intuitiver zu bedienen als die Pflege im System. Weiteres dazu gibt es in Kapitel 4.

## 2.2 Zusammenhang und Komponenten

In Abbildung 1 ist in einer Übersicht der Zusammenhang zwischen den einzelnen Objekten von AGIMENDO.edit beschrieben. Ein Benutzer kann mehreren Views zugeordnet sein, die wiederum jeweils ein oder mehrere Infoobjekte beinhalten. Alle Attribute (Stammdaten und Texte) sind genau einem Infoobjekt zugeordnet. Aus diesen Attributen kann man die Teilmenge bestimmen, die in der Stammdatenpflege zur Bearbeitung in den Views bereitgestellt werden.

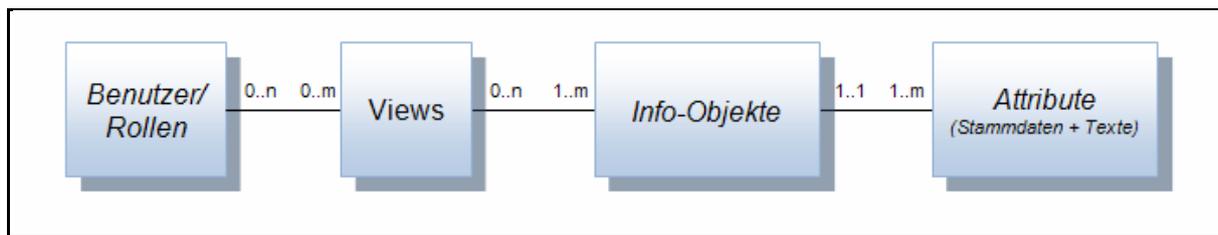


Abbildung 1: Funktionsweise AGIMENDO.edit

AGIMENDO.edit besteht aus drei Komponenten:

- Verwaltung der Views und die Zuordnung von Benutzern bzw. Rollen zu einem View: Web-Dynpro-Anwendung: /ISV/PAQ0DDWDC\_VIEW\_AUTH
- Wizard-basierte Verwaltung der View-Inhalte (Zuordnung von Infoobjekten zu einem bestimmten View und Auswahl der Teilmenge der Attribute) Web-Dynpro-Anwendung: /ISV/PAQ0DDWDC\_WIZARD
- Stammdatenpflege: Web-Dynpro-Anwendung: / ISV/PAQ0DDWDC\_MDM\_MAIN

Die Anwendung wird als Transportauftrag geliefert. Nachdem der Transportauftrag eingespielt wurde, kann die Anwendung als Stand-Alone Variante über eine URL aufgerufen werden. Es besteht auch die Möglichkeit, die Anwendung in einem Portal zu integrieren.

Die URLs des Handbuchs sind allgemein gehalten und müssen an die Ziel-Systemlandschaft angepasst werden.

## 3. Zuordnung von Benutzer bzw. Rollen zu einem View

Die Verwaltung der Views und die Zuordnung von Benutzer bzw. Rollen zu einem View erfolgt über die WebDynpro-Anwendung /ISV/PAQ0DDWDC\_VIEW\_AUTH. Diese ist auf dem System unter dem Pfad [http://<server>:<port>/sap/bc/webdynpro/ISV/PAQ0DDWDC\\_VIEW\\_AUTH](http://<server>:<port>/sap/bc/webdynpro/ISV/PAQ0DDWDC_VIEW_AUTH) zu erreichen.

### 3.1 Neue View anlegen

Über den Startbildschirm der Anwendung ist es möglich, einen View zur Pflege auszuwählen. Dazu kann man in der Suchmaske nach der View Nummer oder der Beschreibung des Views suchen. Die Treffer werden in einer Tabelle unterhalb der Eingabefelder aufgelistet. Wird kein Suchelement eingegeben, werden alle im System existierenden Views aufgelistet.

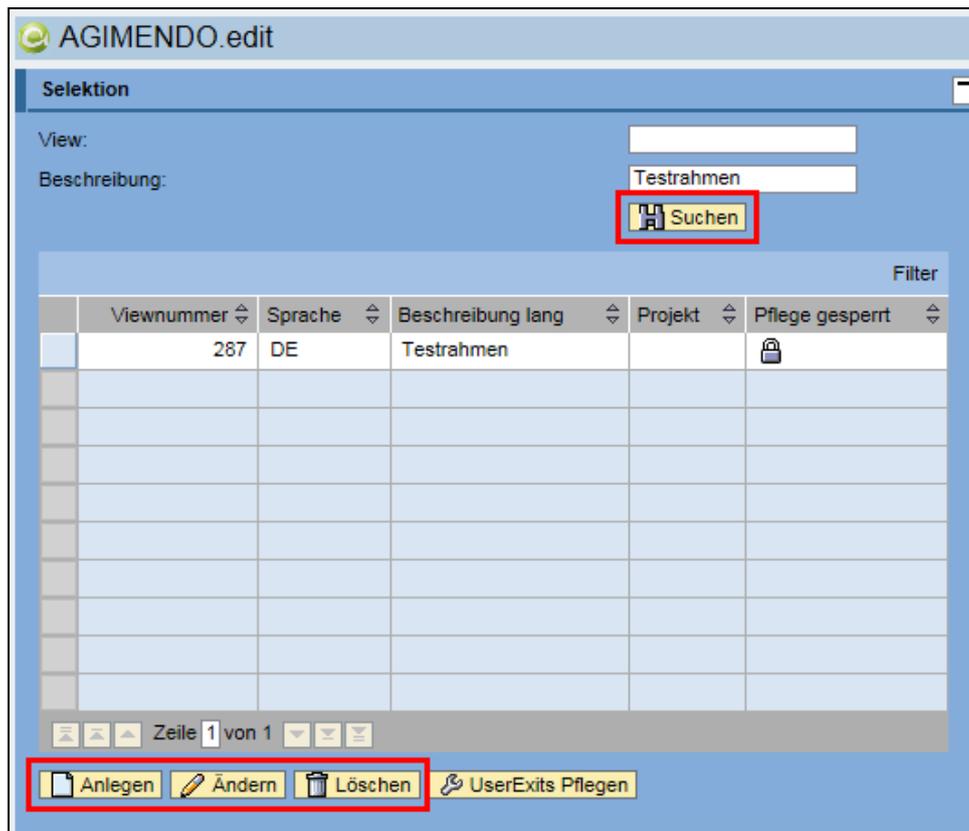


Abbildung 2: Selektion des Views

In Abbildung 2 ist das Ergebnis einer Suche nach dem View „Testrahmen“ dargestellt. Die Liste der Views ist im Standard anhand der View Nummer sortiert. Die Beschreibung des jeweiligen Views und der Name des Projektes, auf das sich der View bezieht, werden in der Anmeldesprache angezeigt. Es wird auch angezeigt, ob der View für die Pflege gesperrt ist. Die Sortierreihenfolge kann durch das Anklicken der Spaltenköpfe beeinflusst werden. Über den Link „Filter“ rechts oben kann eine Filterleiste zur einfachen Suche eingeblendet werden. Unterhalb der Tabelle gibt es die Möglichkeiten, einen neuen View anzulegen, einen vorhandenen View zu ändern bzw. zu löschen.

Bei der Neuanlage von Views müssen die Beschreibung und der Name des Projektes angegeben werden und danach über den Button „Anlegen“ bestätigt werden. Die View Nummer wird automatisch vergeben. Der erstellte View ist nun in der Tabelle aufgelistet und verfügbar.

Beim Ändern eines Views (über den Button „Ändern“) öffnet sich ein Fenster, in dem man die Sprache, die Beschreibung des Views und den Projektnamen ändern kann.

Beim Löschen eines Views wird nach Rückfrage und nach einem Klick auf „Löschen“ der ausgewählte View gelöscht.

### 3.2 Benutzer zu einem View zuordnen

Um einen ausgewählten View zu pflegen, kann über das Anklicken des Views die Pflegansicht angezeigt werden. In der Pflegeansicht gibt es zwei Tabellen. Die rechte Tabelle in Abbildung 3 zeigt alle nicht zugeordneten Benutzer, die linke Tabelle zeigt die zugeordneten Benutzer bzw. Rollen zu dem View. Über die Buttons „Hinzufügen“ und „Entfernen“ besteht die Möglichkeit, Benutzer bzw.

Rollen zu dem View zuzuordnen oder die vorhandenen Zuordnungen zu entfernen. Es besteht die Chance, mehrere Einträge auf einmal auszuwählen und die Operationen dann durchzuführen. Ebenfalls erleichtert die Filterfunktion die Suche nach den Benutzern bzw. Rollen. Ein Speichern der Änderungen ist nicht notwendig.

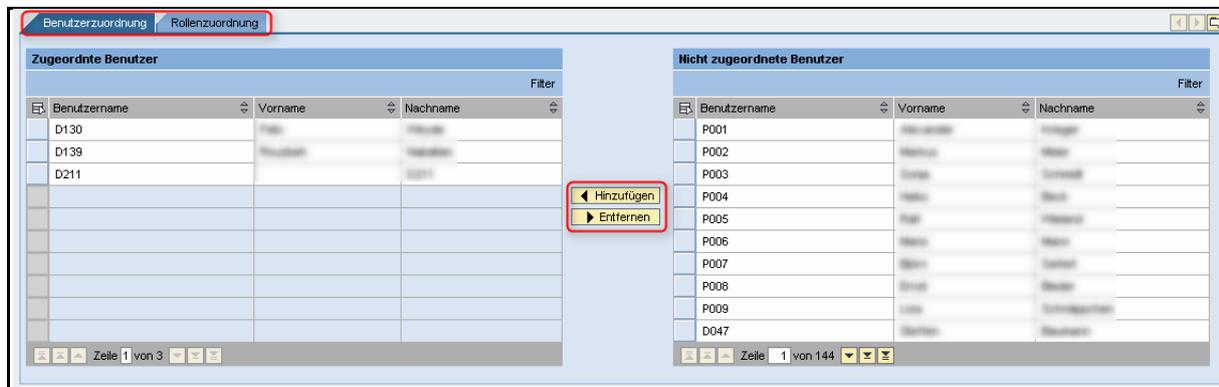


Abbildung 3: Zuordnung von Benutzer bzw. Rollen zu einem View

Später werden alle Benutzer, die in diesem Schritt zu einem bestimmten View zugeordnet wurden, in der Lage sein, entsprechend der Konfigurationen in der „Wizard“-Komponente die Stammdaten der – dem View und damit ihm - zugeordneten Infoobjekte und der ausgewählten Teilmenge von Attributen einzusehen bzw. zu bearbeiten. Diese Konfigurationsmöglichkeiten werden im Folgenden Kapitel beschrieben.

## 4. Wizard-basierte Verwaltung von View Inhalten

Die Zuordnung von Infoobjekten zu einem bestimmten View und die Auswahl der für die Pflege zur Verfügung stehenden Teilmenge der Attribute erfolgt über die WebDynpro-Anwendung / ISV/PAQ0DDWDC\_WIZARD. Diese ist auf dem System unter dem Pfad [http://<server>:<port>/sap/bc/webdynpro/ISV/PAQ0DDWDC\\_WIZARD](http://<server>:<port>/sap/bc/webdynpro/ISV/PAQ0DDWDC_WIZARD) zu erreichen. Im Folgenden wird für „Infoobjekt“ das Wort „Merkmal“ verwendet.

### 4.1 Auswahl eines Views

Über den Startbildschirm der Anwendung („Auswahl eines Views“) ist es möglich, über den Button „Suchen“ einen View auszuwählen. Hier gibt es die Möglichkeit nach einer View Nummer und/oder einer Beschreibung des Views zu suchen.

Die Liste der Views ist im Standard anhand der View Nummer sortiert. Wurden im oberen Bereich bereits Suchkriterien eingegeben, erscheinen in der Liste nur die zutreffenden Views. Falls es keine Suchkriterien gibt, werden alle Views angezeigt. Die Beschreibung des jeweiligen Views und der Name des Projektes, auf das sich der View bezieht, werden in der Anmeldesprache angezeigt. Die Sortierreihenfolge kann durch Anklicken der Spaltenköpfe beeinflusst werden. Über den Link „Filter“ rechts oben kann eine Filterleiste zur einfachen Suche eingeblendet werden. Über die Buttons „Pflege sperren“/ „Pflege erlauben“ kann man einstellen, ob die Stammdaten, die auf dem View

abgebildet werden, neu angelegt/geändert/gelöscht werden können. In der Abbildung 4 ist der Startbildschirm der Wizard-basierten Verwaltung von AGIMENDO.edit dargestellt.

Um den ausgewählten View zu pflegen, kann über den Button „nächster Schritt“ zum zweiten Schritt der Pflegeansicht gesprungen werden.

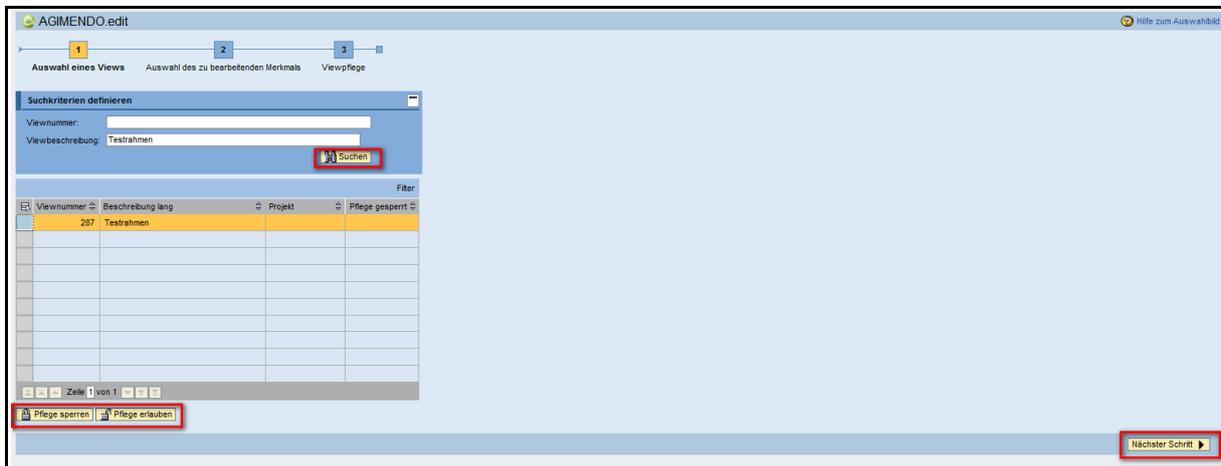


Abbildung 4: Auswahl eines Views

## 4.2 Auswahl des zu bearbeitenden Merkmals

Im zweiten Schritt der Pflegeansicht werden zwei Tabellen dargestellt. Die linke Tabelle zeigt die Merkmale (Infoobjekte) an, die bereits dem in Schritt eins ausgewählten Pflegeview zugeordnet sind. Die rechte Tabelle zeigt die Merkmale an, für die kein Pflegeview definiert wurde und die somit dem View nicht zugeordnet sind.

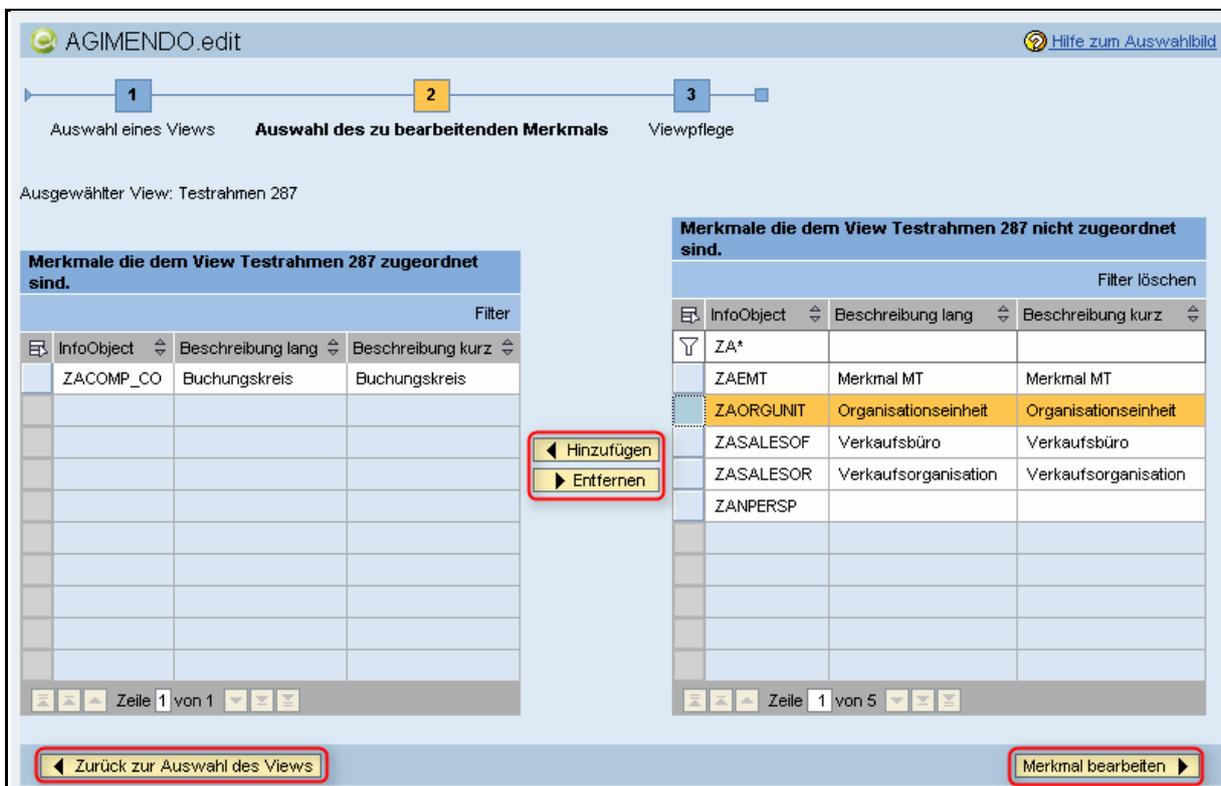
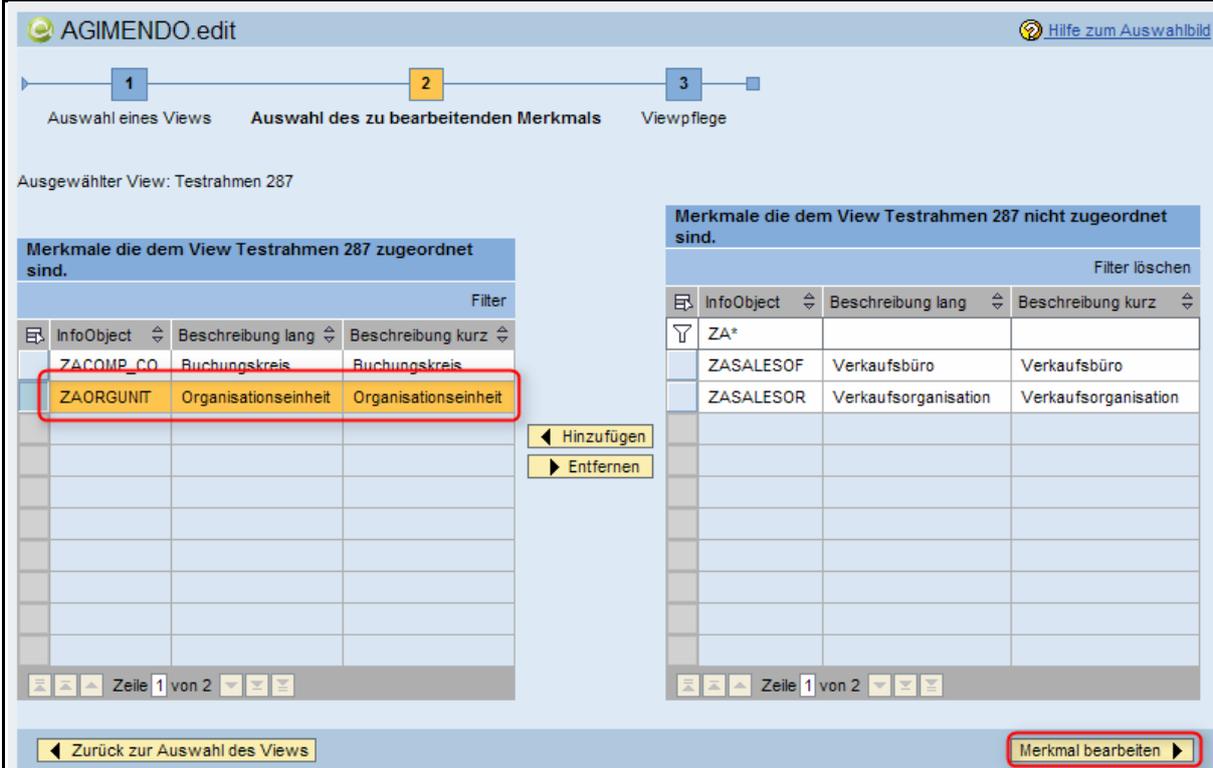


Abbildung 5: Auswahl des zu bearbeitenden Merkmals

Durch Selektieren eines Merkmals in der rechten Tabelle und das Anklicken des Buttons „Hinzufügen“ können dem View weitere Merkmale zugewiesen werden. Für diese Merkmale können im nächsten Schritt Pflegeviews bearbeitet werden. Die Zuordnung eines Infoobjekts zu einem View hat die automatische Zuordnung aller Schlüsselattribute des entsprechenden Infoobjekts zur Folge.

Durch Selektieren eines Merkmals in der linken Tabelle und durch das Anklicken des Buttons „Entfernen“ wird das selektierte Merkmal aus der Liste der Merkmale des ausgewählten Views entfernt und steht nicht mehr zur Bearbeitung zur Verfügung.



AGIMENDO.edit ? Hilfe zum Auswahlbild

1 Auswahl eines Views    2 Auswahl des zu bearbeitenden Merkmals    3 Viewpflege

Ausgewählter View: Testrahmen 287

Merkmale die dem View Testrahmen 287 zugeordnet sind.		
InfoObject	Beschreibung lang	Beschreibung kurz
ZACOMP_CO	Buchungskreis	Buchungskreis
ZAORGUNIT	Organisationseinheit	Organisationseinheit

Filter löschen

InfoObject	Beschreibung lang	Beschreibung kurz
ZA*		
ZASALESOF	Verkaufsbüro	Verkaufsbüro
ZASALESOR	Verkaufsorganisation	Verkaufsorganisation

◀ Hinzufügen    ▶ Entfernen

Zeile 1 von 2    Zeile 1 von 2

◀ Zurück zur Auswahl des Views    Merkmal bearbeiten ▶

Abbildung 6: Ausgewähltes Merkmal bearbeiten

Durch Selektieren eines Merkmals in der linken Tabelle können bereits vorhandene Pflegeviews durch Klicken auf den Button „Merkmal bearbeiten“ im dritten Schritt bearbeitet werden.

Über den Button „Zurück zur Auswahl des Views“ kann zu dem ersten Schritt der Pflegeansicht gesprungen werden.

**Hinweis:** Die Merkmalseinstellungen können abweichen, falls das Merkmal zu verschiedenen Views zugeordnet wurde. Es ist also möglich, in einem View bspw. zwei Felder zu definieren und in einem anderen View drei andere Attribute des Merkmals zu pflegen.

### 4.3 Viewpflege

Im dritten Schritt wird dem Pflegeview für den in Schritt 1 ausgewählten View und das in Schritt 2 ausgewählte Merkmal definiert.

Abbildung 7 zeigt die Pflegeansicht in der eine Teilmenge der Attribute für das selektierte InfoObjekt ausgewählt und konfiguriert werden kann. Im unteren Bereich der Seite gibt es zwei Tabellen. In der

linken Tabelle werden die bereits zur Pflege zugeordneten Felder angezeigt. In der rechten Tabelle werden die Felder, die nicht zur Pflege zugeordnet wurden angezeigt.

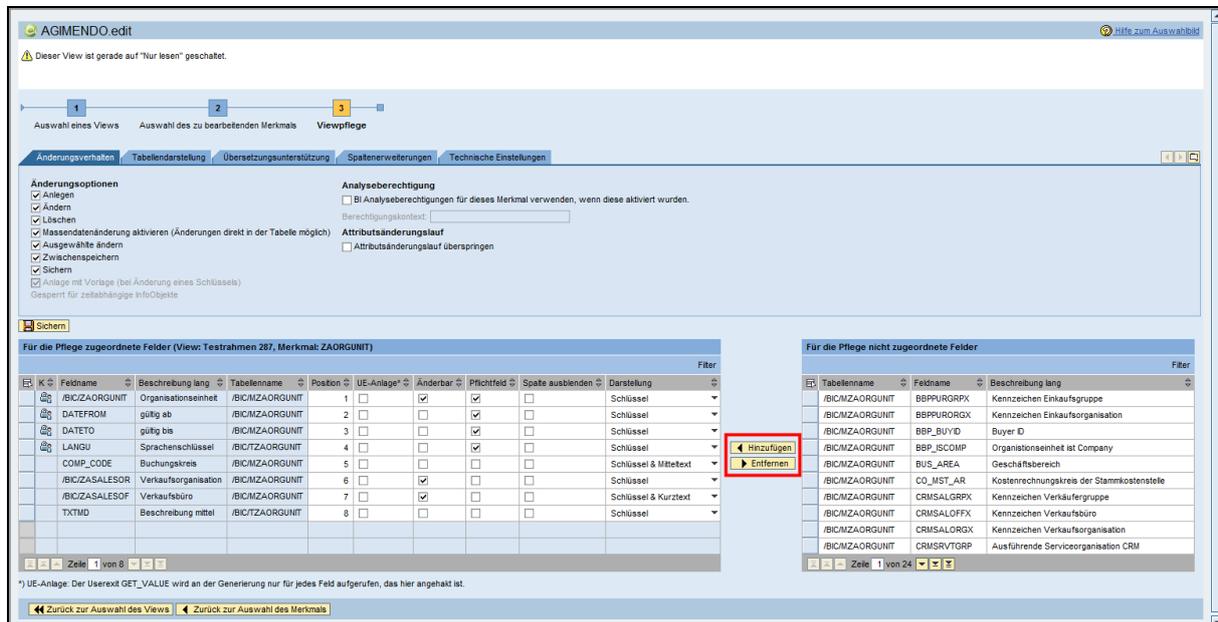


Abbildung 7: Attribute zuordnen

Durch Selektieren eines Feldes in der rechten Tabelle und durch anschließenden Klick auf den Button „Hinzufügen“ wird das selektierte Feld dem Pflegeview hinzugefügt. Durch Selektieren eines Feldes in der linken Tabelle und durch anschließenden Klick auf den Button „Entfernen“ wird das selektierte Feld aus dem Pflegeview entfernt. Die Reihenfolge der zugeordneten Attribute kann durch die Knöpfe „Nach Oben“ und „Nach Unten“ gesteuert werden.

**Hinweis:** Alle Schlüsselfelder werden automatisch hinzugefügt. Ist ein Attribut bspw. zeitabhängig oder mehrsprachig, so wird das zugehörige Schlüsselfeld (LANGU, DATEFROM, DATETO, etc.) automatisch beim Hinzufügen oder Entfernen des Attributs mittransportiert.

Durch Klick auf den Button „Auswahl des Merkmals“ gelangt man zurück zum zweiten Schritt. Durch Klick auf den Button „Auswahl des Views“ gelangt man zurück zum ersten Schritt.

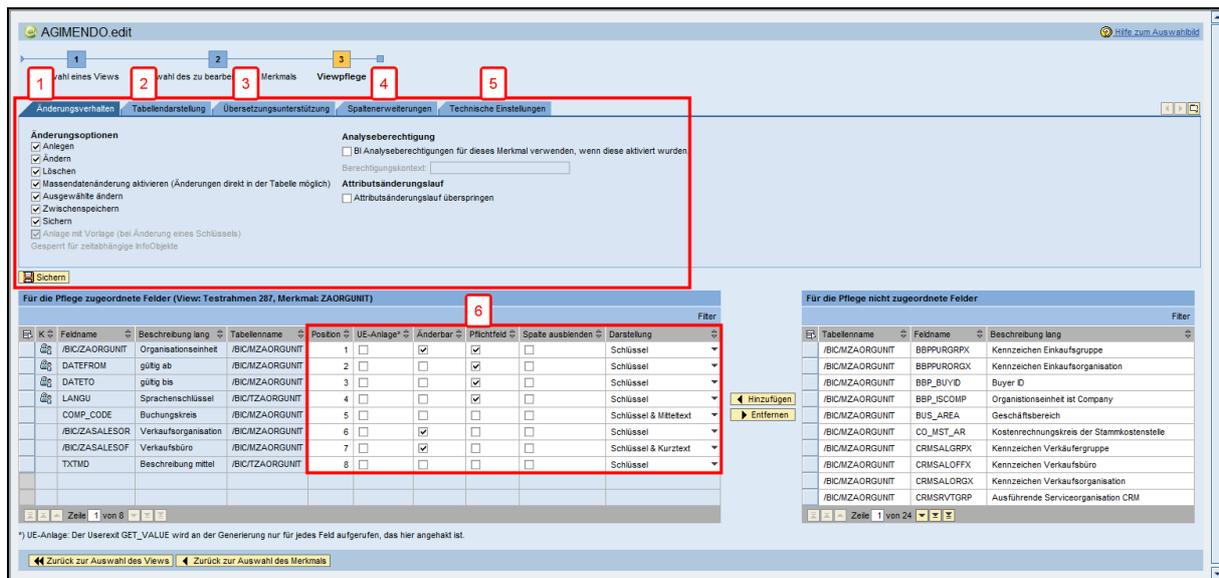


Abbildung 8: Viewpflege

Wie in Abbildung 8 demonstriert wird, gibt es sechs Bereiche, die jeweils die Darstellung und den Funktionsumfang der tatsächlichen Stammdatenpflegemaske steuern. Die folgenden Punkte beschreiben diese Konfigurationsmöglichkeiten in der Reihenfolge der Nummerierung.

### 4.3.1 Änderungsverhalten

Das Änderungsverhalten bestimmt die Möglichkeiten, die im Pflegemodus bereit gestellt werden. Über verschiedene Checkboxes können verschiedene Einstellungen bestimmt werden.

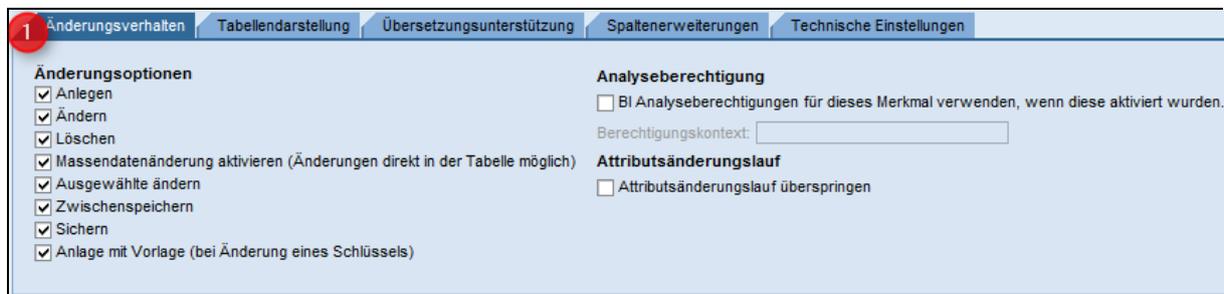


Abbildung 9: Konfiguration des Pflegeviews (Änderungsverhalten)

- Anlegen:** Bestimmt, ob die Anwender neue Datensätze anlegen dürfen.
- Ändern:** Bestimmt, ob die Anwender vorhandene Stammdaten ändern dürfen.
- Löschen:** Bestimmt, ob die Anwender vorhandene Stammdaten löschen dürfen.
- Massendatenänderung:** Ist dieses Feld selektiert, können später Änderungen direkt in der Tabelle durchgeführt werden, sowie mehrere Stammdaten gleichzeitig ausgewählt und für alle dieselben Änderungen an Nicht-Schlüsselattributen vorgenommen werden.

<i>Ausgewählte ändern:</i>	Bestimmt, ob die Anwender die Möglichkeit haben sollen, mehrere ausgewählte Datensätze gleichzeitig zu ändern (nur mit aktivierter Massenänderung möglich).
<i>Zwischenspeichern:</i>	Bestimmt, ob die Anwender Änderungen zwischenspeichern dürfen (ohne neues Laden und ohne starten des Änderungslaufes, nur mit aktivierter Massenänderung möglich).
<i>Sichern:</i>	Bestimmt, Speichern dürfen (inklusive dem Starten des Attributänderungslaufes sowie dem anschließenden Neuladen aller Daten, nur mit aktivierter Massenänderung möglich)
<i>Anlage mit Vorlage:</i>	Ist dieses Feld selektiert, wird bei der Änderung eines Schlüsselattributes der Wert nicht überschrieben, sondern ein neuer Datensatz mit diesem neuen Wert angelegt. Dieses Feld ist standardmäßig ausgewählt und kann nur abgewählt werden, wenn das Feld Massendatenänderung angehakt ist. Dieses Feld ist für zeitabhängige InfoObjekte automatisch ausgewählt .
<i>BI Analyseberechtigung:</i>	Ist dieses Feld selektiert, wird die BI-Analyseberechtigung für die Endanwender geprüft. Es filtert die verfügbaren Daten entsprechend den Berechtigungen, so dass der Endanwender nur gemäß der BI Analyseberechtigungen pflegen darf. Bei der Selektion dieser Option wird das Eingabefeld <i>Berechtigungskontext</i> eingabebereit. Wenn hier ein InfoCube als InfoProvider eingegeben wird, werden die Berechtigungen geprüft, die sich nur auf diesen InfoProvider beziehen.

#### *Attributsänderungslauf überspringen*

Ist dieses Feld aktiviert, werden alle Attributsänderungsläufe in alle Fälle (Speichern, Löschen, Anlegen Massendatenänderungsmodus oder nicht) übersprungen.

#### *Alle bearbeiteten Zeilen gleichzeitig speichern um den Durchsatz zu erhöhen*

Ist dieses Feld aktiviert, werden alle Änderungen zusammengefasst und gleichzeitig gespeichert. Im Gegensatz zum einzelnen Speichern, bei dem jede Änderung nach einander gespeichert wird. Das Speicherverhalten ändert sich für den User nicht. Es ist lediglich eine Performance Verbesserung.

### 4.3.2 Tabellendarstellung

Dieses Tab bezieht sich auf die Darstellung der Stammdatentabelle. Die Anzahl der angezeigten Zeilen, die Anzahl der insgesamt angezeigten Spalten, sowie die Anzahl der fixierten Spalten können

hier konfiguriert werden. Außerdem kann man die Filterzeile ein- und ausblenden und neue Texte für Buttons definieren, die die Standardtexte überschreiben (man kann auch nicht die Texte direkt eingeben, sondern ihre OTR-Aliase – dafür wird eine entsprechende Checkbox ausgewählt).

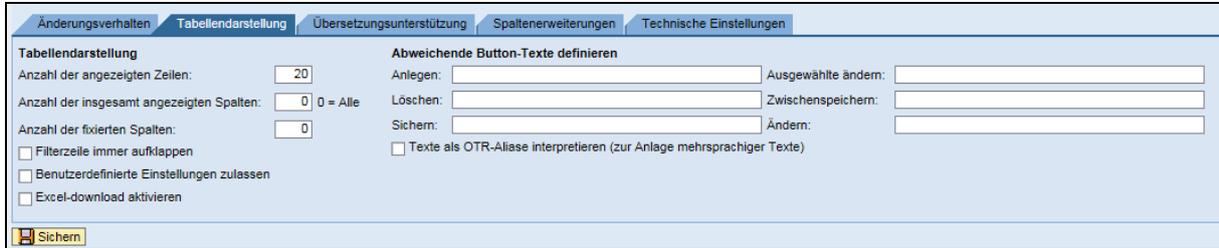


Abbildung 10: Konfiguration des Pflegeviews (Tabellendarstellung)

**Hinweis:** Ab der SAP-BW Version 7.4 hat die SAP die UI-Richtlinien geändert. Durch diese Richtlinie wurde die bekannte Filterzeile als deprecated markiert, sollte also nicht mehr verwendet werden. Bisher kann über einen URL-Parameter (WDUIGUIDELINE=GL11) das alte Verhalten wiederhergestellt werden. Alternativ kann auch das vollständige System wieder auf die „alte“ UI-Richtlinie umgestellt werden. Kontaktieren Sie hierzu bitte Ihre SAP Basis.

Die zweite Option („Benutzerdefinierte Einstellungen zulassen“) ermöglicht es dem Benutzer der Pflegeanwendung eigene Einstellungen am View (Varianten) vorzunehmen. Dabei werden alle Varianten für ungültig erklärt und gelöscht, sobald Änderungen am View (eine neue Version im Wizard gespeichert wird) vorgenommen werden.

Die letzte Option gibt dem Benutzer der Pflegeanwendung die Möglichkeit, den Inhalt der Tabelle in eine Excel Datei zu exportieren.

**Hinweis:** Alle Änderungen bezogen auf die Tabellendarstellung werden erst nach dem Speichern über den Button „Sichern“ wirksam.

### 4.3.3 Übersetzungsunterstützung

Das dritte Tab bezieht sich auf die Übersetzungsunterstützung. Dieses Tab ist nur für Infoobjekte mit sprachabhängigen Texten verfügbar. Hier kann eingestellt werden, für welche Sprachen Datensätze (vor-)angelegt werden sollen und ob anderssprachige Datensätze angezeigt werden sollen oder nicht. Über die Buttons „Zeile anhängen“ bzw. „Zeile einfügen“ können neue Einträge zu der Liste vor bzw. nach einem Eintrag hinzugefügt werden. Die vorhandenen Einträge können durch den Button „Zeile löschen“ aus der Liste entfernt werden. Über die Spalte Reihenfolge kann die Reihenfolge der angezeigten Datensätze in der Stammdatenpflege gesteuert werden. Über die Auswahl einer Checkbox in der Spalte „Sprache ausblenden“ kann man einstellen, dass die Datensätze mit Texten in dieser Sprache in der Stammdatenpflege nicht angezeigt werden.

**Andere Sprachen ausblenden:** Ist dieses Feld selektiert, werden nur für die in der Tabelle sichtbar definierten Sprachen Datensätze in der Stammdatenpflege angezeigt.

**Texte bei Neuanlagen alle (konfigurierten) Sprachen kopieren:**

Ist dieses Feld selektiert, werden die Texte bei Neuanlage aus dem Textfeld für die aktuelle Sprache in die Textfelder dieses Datensatzes für alle weiteren Sprachen kopiert.

*Textänderungen an UserExit weiterleiten:*

Bei der Auswahl dieser Checkbox werden die Änderungen in den Textfeldern an den UserExit „TEXT\_CHANGED“ (implementiert in der Methode „TEXT\_CHANGED“ der Klasse /ISV/PAQ0DCX\_UE\_ABSTRACT) weitergeleitet.

**Hinweis:** Die Pflege mehrerer Sprachen in dieser Tabelle hat zur Folge, dass beim nächsten Aufruf des Views in der Stammdatenpflege Datensätze in diesen Sprachen bereits mit angelegt werden. Löschen von Einträgen in dieser Tabelle hat aber **nicht** das Löschen von zuvor angelegten Datensätzen zur Folge.

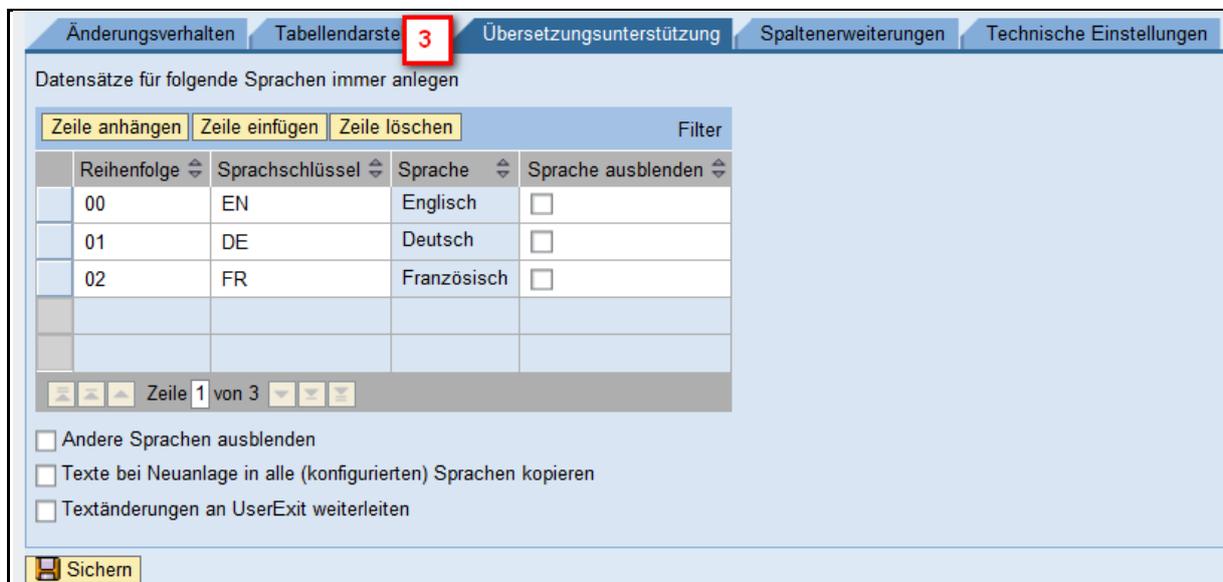


Abbildung 11: Konfiguration des Pflegeviews (Übersetzungsunterstützung)

**Hinweis:** Alle Änderungen bezogen auf die Übersetzungsunterstützung werden erst nach dem Speichern über den Button „Sichern“ wirksam.

#### 4.3.4 Spaltenerweiterungen.

Im vierten Tab kann man Parameter für die Anzeige zusätzlicher Attribute und/oder Texte zu den InfoObjects eingeben. Um zusätzliche Attribute/Texte zu den InfoObjects in der Stammdatenpflege einzublenden, stehen die beiden folgenden UserExit-Klassen zur Verfügung:

- ▶ /ISV/CL\_AE\_EXT\_COL\_IOBJ\_ATTR
- ▶ /ISV/CL\_AE\_EXT\_COL\_IOBJ\_TXT

Die UserExit-Klasse /ISV/CL\_AE\_EXT\_COL\_IOBJ\_ATTR dient dazu, Attribute von InfoObjects einzublenden. Zum Beispiel soll in der Tabelle Produkt, in der das Attribut Werk angezeigt wird, auch das Land des Werks angezeigt werden.

Der UserExit /ISV/CL\_AE\_EXT\_COL\_IOBJ\_TXT dient dazu, Texte von Attributen für InfoObjects einzublenden. Zum Beispiel soll zusätzlich zum Länderschlüssel auch der Kurztext für das Land angezeigt werden.

In die Tabelle werden folgende Parameter eingetragen:

<i>Position:</i>	Stelle des eingeblendeten Attributs/ Textes in dem Stammdatenpflegeview.
<i>Feld ID:</i>	Technischer Name für das Feld des eingeblendeten Attributs/Textes in dem Stammdatenpflegeview.
<i>Spalten ausblenden</i>	<i>Checkbox für die Ein- / Ausblendung der Spalte</i>
<i>Aktiv:</i>	<i>Checkbox für die Aktivierung/ Deaktivierung der Parameter</i>
<i>Gepuffert:</i>	<i>Einstellung, ob der über die UserExit Klasse ermittelte Text zu dem Attribut während der Laufzeit gepuffert wird</i>
<i>User-Exit-Klasse:</i>	User Exit Klasse: /ISV/CL_AE_EXT_COL_IOBJ_TXT für einen zusätzlichen Text oder /ISV/CL_AE_EXT_COL_IOBJ_ATTR für eine zusätzliche Attribut.
<i>Parameter (1):</i>	das InfoObject (Merkmal), zu dem das Attribut/der Text angezeigt werden soll
<i>Parameter (2):</i>	Name des Attributs, das angezeigt werden muss/ deren Text angezeigt werden soll
<i>Parameter (3):</i>	für zusätzlich eingeblendeten Text: Eingab vom Texttyp –TXTSH für Kurztext, TXTMD für mittellangen Text, oder TXTLG für Langtext.

Änderungsverhalten | Tabellendarstellung | Übersetzungsunterstützung | **4** Spaltenerweiterungen | Technische Einstellungen

Datentabelle um folgende User-Exit-Spalten erweitern.

Zeile anhängen | Zeile löschen | Filter

Position	Feld-ID	Aktiv	Spalte ausblenden	Gepuffert	User-Exit-Klasse	Parameter	Parameter	Parameter
6	Z_ZAETI004	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	/ISV/CL_AE_EXT_COL_IOBJ_ATTR	ZAETI002	ZAETI004	Parameter
7	Z_ZAETI005	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	/ISV/CL_AE_EXT_COL_IOBJ_ATTR	ZAETI003	ZAETI004	
8	Z_07	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	/ISV/CL_AE_EXT_COL_IOBJ_TXT	ZAETI002	ZAETI004	TXTSH
9	Z_06	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	/ISV/CL_AE_EXT_COL_IOBJ_TXT	ZAETI003	ZAETI004	TXTSH
10	Z_08	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	/ISV/CL_AE_EXT_COL_IOBJ_TXT	ZAETI002	ZAETI005	TXTSH
11	Z_09	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	/ISV/CL_AE_EXT_COL_IOBJ_TXT	ZAETI003	ZAETI005	TXTSH

Sichern

Für die Pflege zugeordnete Felder (View: 430, Merkmal: ZAETI001)

K	Feldname	Beschreibung lang	Tabellename	Position	UE-Anlage*	Änderbar	Pflichtfeld	Spalte ausblenden	Darstellung
	/BIC/ZAETI001		/BIC/MZAETI001	1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	LANGU	Sprachenschlüssel	/BIC/TZAETI001	2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	TXTSH	Beschreibung kurz	/BIC/TZAETI001	3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	/BIC/ZAETI003		/BIC/MZAETI001	4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	/BIC/ZAETI002		/BIC/MZAETI001	5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Schlüssel

InfoObject ZAETI003 kommt im Merkmal ZAETI001 als Attribut mit Schlüssel /BIC/ZAETI003

Attribut ZAETI004 des InfoObjects ZAETI003 und sein Textfeld sollen eingblendet werden

Texttyp für das Textfeld des Attributs ZAETI003

Abbildung 12: Konfiguration des Pflegeviews (Spaltenerweiterungen)

**Hinweis:** Es ist möglich eigene Spaltenerweiterungen zu implementieren, die die angezeigten Informationen aus ganz anderen Quellen ermitteln können. Dazu muss eine Redefinition der Klasse /ISV/CL\_AE\_EXT\_COL\_IOBJ\_ATTR angelegt werden und dann als User-Exit-Klasse verwendet werden. Die an der Oberfläche erfassten Parameter stehen dann in den Methoden zur Verfügung.

#### 4.3.5 Technische Einstellungen.

Im fünften Tab kann man die Einstellung für die korrekte Darstellung von Spezialwährungen mit abweichender Anzahl von Dezimalstellen (z.B. Hungarian Forint (HUF), Japanese Yen (JPY), Korean Won (KRW), Chilean Peso (CLP) etc.) aktivieren. Diese Option ist nur notwendig, wenn Kennzahlen mit eben diesen Währungen im View gepflegt werden. Aus Performance-Gründen ist diese Option in der Voreinstellung abgeschaltet.

Die zweite Option („Bei Sichern, Neu anlegen und Löschen Tabellenstatus beibehalten“) bewirkt, dass nach einem neu Laden der Datentabelle der Status der Tabelle beibehalten wird. Der Filter bleibt gesetzt und die Scrollposition bleibt ebenfalls bestehen.

Der zweite Bereich in diesem Tab stellt eine URL-Vorlage zum aktuell gewählten View/InfoObjekt dar. Ein Aufruf der Stammdatenpflege mit dieser URL-Vorlage als zusätzlicher Parameter ermöglicht ein Filtern direkt auf das gewählten View/InfoObjekt-Paar. Über den Link „Link testen“ kann direkt zum aktuellen View/InfoObjektpaar gesprochen werden.

**Hinweis:** Hierbei wird ein neues Fenster mit dem direkten Link zur Pflegeanwendung auf dem ABAP-Stack geöffnet.

Der Dritte Bereich stellt die Optionen des Transportmanagements dar. Siehe hierfür Kapitel 8.

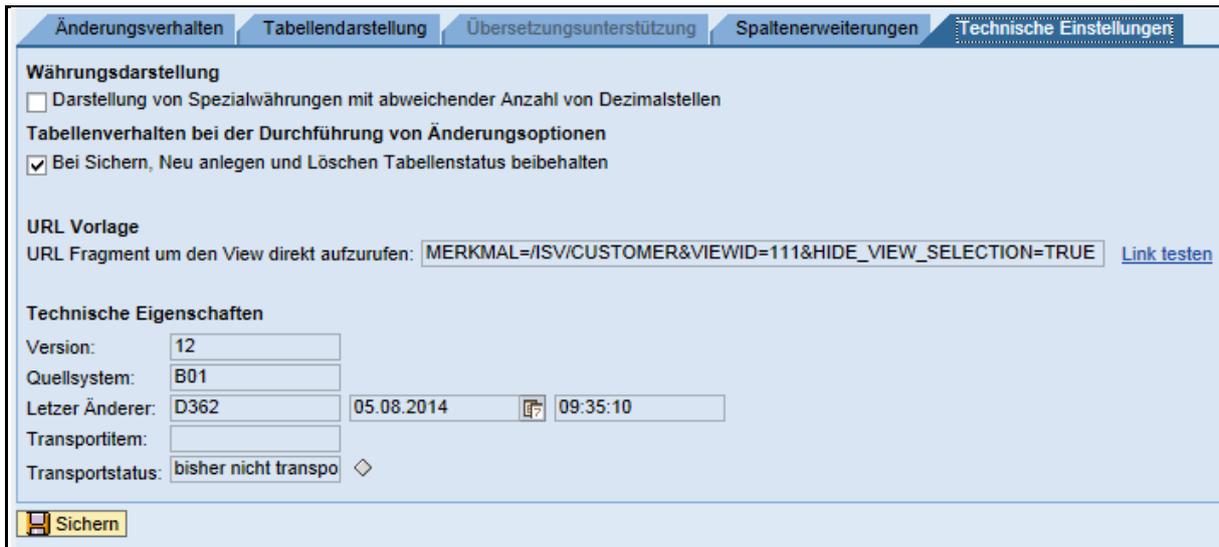
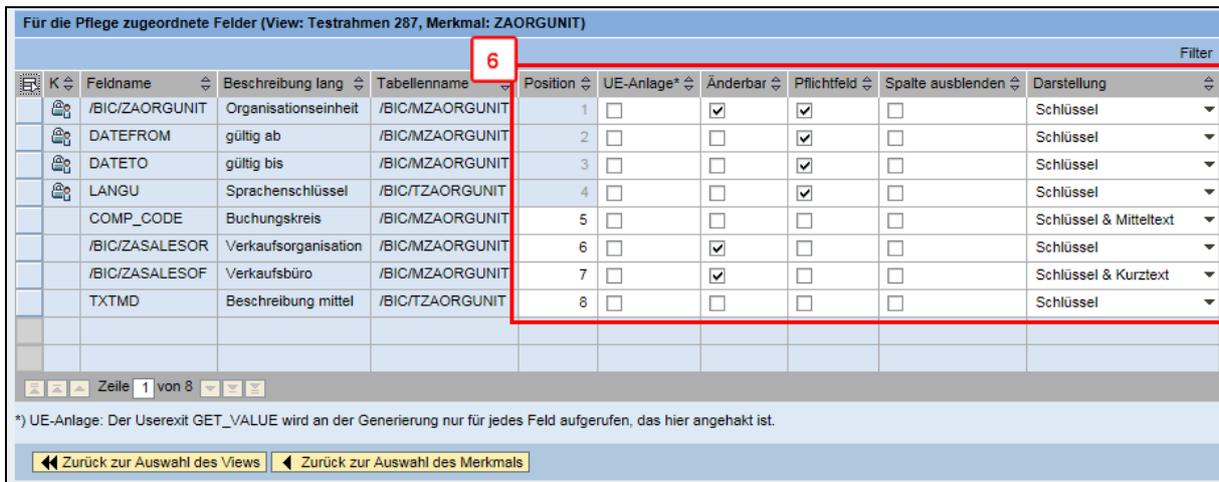


Abbildung 13: Konfiguration des Pflegeviews (Technische Einstellungen)

#### 4.3.6 Konfiguration des für die Pflege zugeordneten Attributs

In der linken Tabelle des unteren Bereichs der Pflegeoberfläche besteht die Möglichkeit, die Darstellung und das Änderungsverhalten der bereits zugeordneten Attribute zu steuern.



K	Feldname	Beschreibung lang	Tabellenname	Position	UE-Anlage*	Änderbar	Pflichtfeld	Spalte ausblenden	Darstellung
	/BIC/ZAORGUNIT	Organisationseinheit	/BIC/MZAORGUNIT	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	DATEFROM	gültig ab	/BIC/MZAORGUNIT	2	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	DATETO	gültig bis	/BIC/MZAORGUNIT	3	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	LANGU	Sprachenschlüssel	/BIC/ITZAORGUNIT	4	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	COMP_CODE	Buchungskreis	/BIC/MZAORGUNIT	5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Schlüssel & Mitteltext
	/BIC/ZASALESOR	Verkaufsorganisation	/BIC/MZAORGUNIT	6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Schlüssel
	/BIC/ZASALESOF	Verkaufsbüro	/BIC/MZAORGUNIT	7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Schlüssel & Kurztext
	TXTMD	Beschreibung mittel	/BIC/ITZAORGUNIT	8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Schlüssel

Abbildung 14: Konfiguration des Pflegeviews (Attribute definieren)

In diesem Bereich können je Feld sechs Einstellungen festgelegt werden:

- Position:** Positionierung der Spalte im View. Im View wird zur Laufzeit die Reihenfolge der Spalten nach dem hier angegebenen Wert aufsteigend sortiert.
- UE-Anlage:** Bei einer Neuanlage von Stammdaten ist das betreffende Feld bereits vorbelegt (s. dazu auch Kapitel 6.4.2)
- Änderbar:** Stammdaten sind eingabebereit und können direkt in der Tabelle bearbeitet werden.
- Pflichtfeld:** Damit kann man definieren, welche Felder bei der Anlage bzw. beim Ändern nicht leer gelassen werden dürfen. Schlüsselfelder sind Pflichtfelder, die anderen Attribute können es sein.

**Spalte ausblenden:** Alle Spalten, die hier mit einem Haken versehen werden, werden im View ausgeblendet, sind aber technisch im View vorhanden (und beispielsweise über einen UserExit auslesbar/änderbar)

**Darstellung:** Bietet die Möglichkeit zusätzlich zu dem Schlüssel einer Merkmalsausprägung auch der Text bzw. nur den Text anzuzeigen.

**Hinweise:** Die verschiedenen Darstellungsoptionen für die Attribute werden dynamisch ermittelt und in einer Dropdown-Liste zur Verfügung gestellt. Dabei sind folgende Punkte zu beachten:

*Wird ein Attribut dem Pflegeview zugeordnet, wird der Defaultwert der Darstellung auf „Schlüssel“ gesetzt.*

*Weitere Optionen zur Darstellung werden nur dann angezeigt, falls das Merkmal die entsprechenden Textattribute besitzt.*

*Bei einem Schlüsselattribut zu einem bestimmten Merkmal besteht **nicht** die Möglichkeit zusätzliche Texte anzuzeigen. Dafür können die entsprechenden Textattribute aus der rechten Tabelle dem Pflegeview zugeordnet werden.*

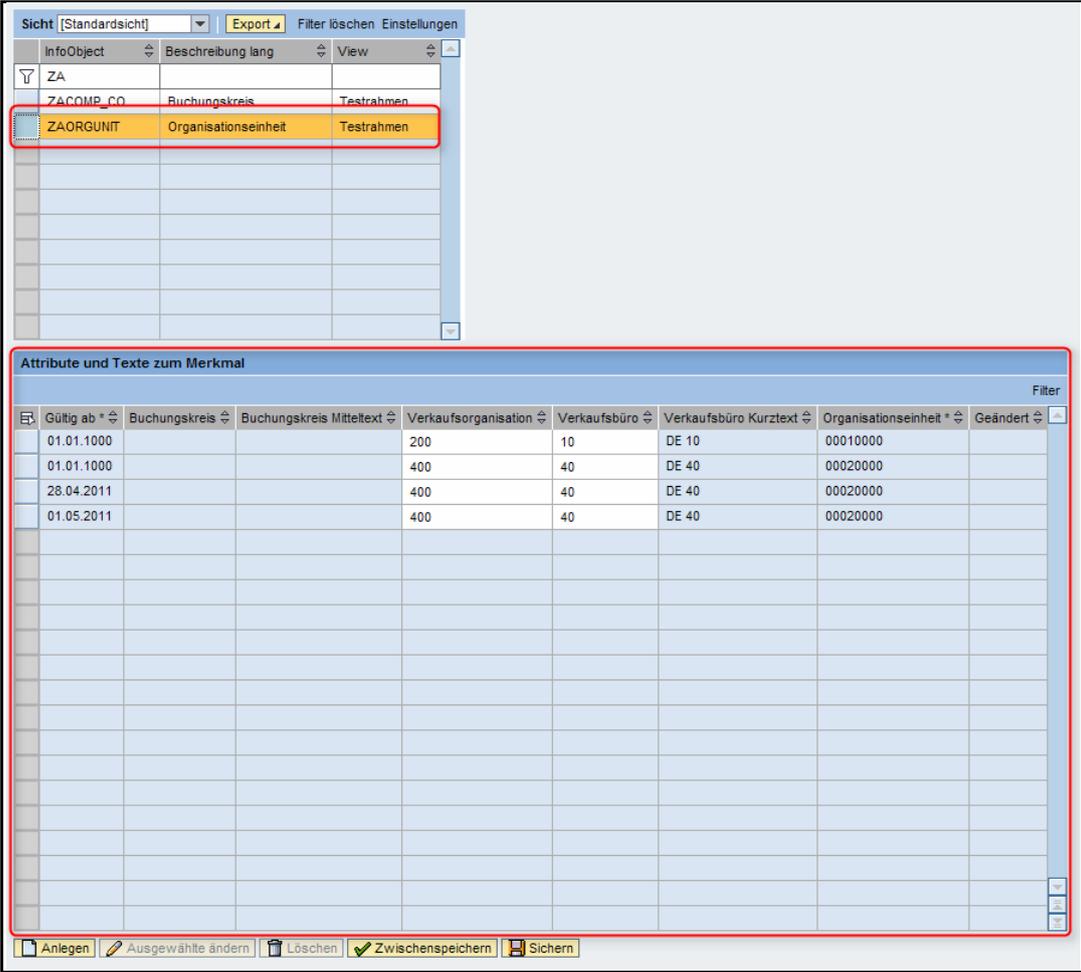
*Wird ein Attribut als änderbar gekennzeichnet, wird automatisch der Schlüssel zu der zuvor ausgewählten Option hinzugefügt.*

## 5. Stammdatenpflege

Die eigentliche Pflege der Stammdaten erfolgt über die WebDynpro-Anwendung /ISV/PAQ0DDWDC\_MDM\_MAIN. Diese ist auf dem System unter dem Pfad [http://<server>:<port>/sap/bc/webdynpro/ISV/PAQ0DDWDC\\_MDM\\_MAIN](http://<server>:<port>/sap/bc/webdynpro/ISV/PAQ0DDWDC_MDM_MAIN) zu erreichen.

### 5.1 Aufruf der Stammdatenpflege

Über den Startbildschirm der Anwendung ist es möglich, ein Merkmal aus der oberen Tabelle auszuwählen. Die daraufhin erscheinende Tabelle wird entsprechend dem Änderungsverhalten (Buttons) und der Darstellung der einzelnen Attribute, die im Wizard bestimmt wurden, angezeigt. Der Benutzer kann die Stammdaten des Merkmals nun anhand dieser Konfigurationen einsehen bzw. bearbeiten.



The screenshot displays the 'Stammdatenpflege' (Master Data Maintenance) interface. At the top, there is a navigation bar with 'Sicht [Standardsicht]', 'Export', and 'Filter löschen Einstellungen'. Below this is a table of views:

InfoObject	Beschreibung lang	View
ZA		
ZACOMP_CO	Buchungskreis	Testrahmen
ZAORGUNIT	Organisationseinheit	Testrahmen

The selected view 'ZAORGUNIT' is highlighted with a red border. Below this is a section titled 'Attribute und Texte zum Merkmal' with a 'Filter' button. It contains a table with the following data:

Gültig ab *	Buchungskreis	Buchungskreis Mitteltext	Verkaufsorganisation	Verkaufsbüro	Verkaufsbüro Kurztext	Organisationseinheit *	Geändert
01.01.1000			200	10	DE 10	00010000	
01.01.1000			400	40	DE 40	00020000	
28.04.2011			400	40	DE 40	00020000	
01.05.2011			400	40	DE 40	00020000	

At the bottom of the interface, there is a toolbar with buttons: 'Anlegen', 'Ausgewählte ändern', 'Löschen', 'Zwischenspeichern', and 'Sichern'.

Abbildung 15: Startbildschirm Stammdatenpflege

**Hinweis:** Der Anwender hat nur Zugriff auf die Views, die ihm im Rahmen des Schritts „Benutzer zu einem View zuordnen“ (Kapitel 3.2) zugeordnet wurden

Folgende Abbildung veranschaulicht den Zusammenhang zwischen den im Wizard vorgenommenen Einstellungen und deren Wirkung auf die Pflegeoberfläche der Stammdatenpflege.

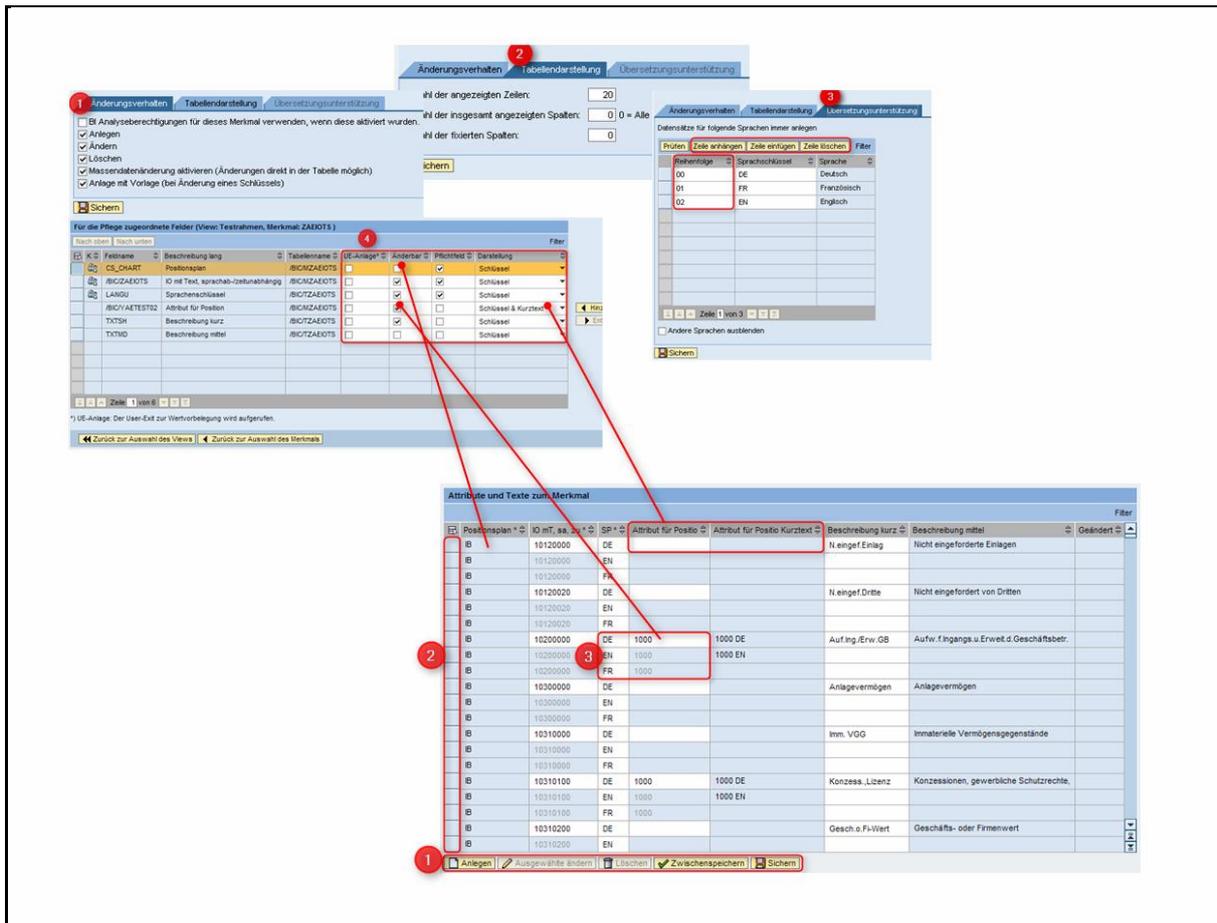


Abbildung 16: Wirkung der im Wizard vorgenommenen Einstellungen auf die Pflegeoberfläche

## 5.2 Parametrisierter Aufruf der Stammdatenpflege

Es ist möglich, die Stammdatenpflege für ein bestimmtes View und ein Merkmal parametrisiert aufzurufen. Dabei werden die View Nummer und der Name des Merkmals via URL-Parameter der Anwendung übergeben. Ungültige Aufrufe (User hat keine Berechtigung, Kombination existiert nicht, etc.) werden erkannt und ignoriert.

Beispiel: Der Aufruf der URL

[http://<server>:<port>/sap/bc/webdynpro/isv/paq0ddwdc\\_mdm\\_main?VIEWID=<ViewNummer>&MERKMAL=<Merkmalsname>](http://<server>:<port>/sap/bc/webdynpro/isv/paq0ddwdc_mdm_main?VIEWID=<ViewNummer>&MERKMAL=<Merkmalsname>) führt zu Auswahl der Stammdatenpflege für den entsprechenden View und das Merkmal.



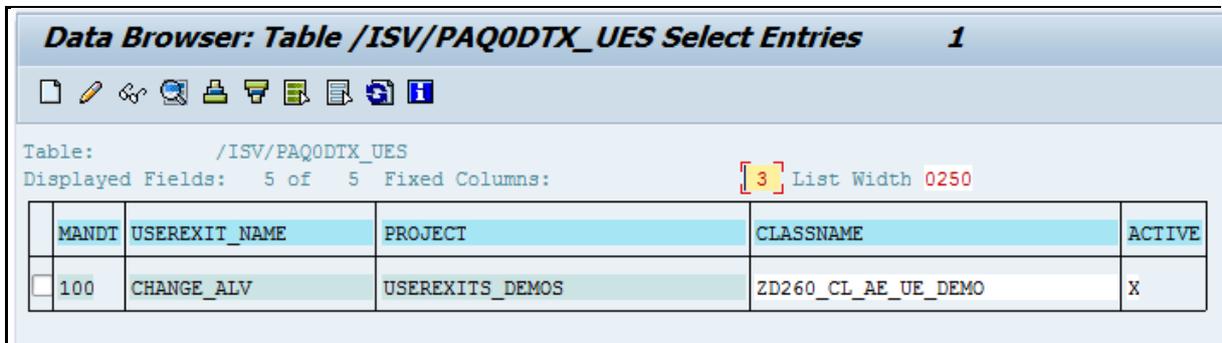
## 6. AGIMENDO.edit User Exits

Eine Stärke von AGIMENDO.edit ist die Erweiterbarkeit und vielfältigen Möglichkeiten, um die „einfache“ Pflege von Stammdaten durch User Exits zu ergänzen. In diesem Kapitel werden diese Möglichkeiten dargestellt und erklärt.

### 6.1 Definition und Ablage von User Exits

Die Bereitstellung der User Exits erfolgt in Form definierter Methodenaufrufe, die durch kundeneigene Implementierungen überschrieben werden können. Dazu ist eine Klasse zu erstellen, die in der Vererbungshierarchie unter der abstrakten Klasse /ISV/PAQ0DCX\_UE\_ABSTRACT (Paket /ISV/PEX\_AMA) steht. Die Realisierung erfolgt durch die Redefinition der Methoden in der abgeleiteten Klasse. Alle projektbezogenen UserExits und ihnen entsprechende Klassen werden in die Datenbanktabelle /ISV/PAQ0DTX\_UES eingetragen (Abbildung 18). Um einen User Exit zu aktivieren, muss das Flag „ACTIVE“ in der Tabelle /ISV/PAQ0DTX\_UES gesetzt sein. Die Pflege der Tabelle kann entweder direkt erfolgen (SE16) oder über die Oberflächen von AGIMENDO.edit (siehe 6.2). Technisch ist es notwendig, dass es von der abstrakten Klasse /ISV/PAQ0DCX\_UE\_ABSTRACT erweitert werden muss, es wird aber **dringend empfohlen, von der Default-Implementierung /ISV/PAQ0DCX\_UE\_IMP** zu erben. In dieser Klasse stehen erweiterte Utility-Methoden zur Bearbeitung zur Verfügung. Außerdem kann so auf die allgemeine Funktionalität zugegriffen werden.

Beispielhaft ist im folgenden Screenshot die Konfiguration des UserExits CUSTOM\_VALIDATION durch die Klasse ZD260\_CL\_AE\_UE\_DEMO dargestellt (Abbildung 18):



**Data Browser: Table /ISV/PAQ0DTX\_UES Select Entries 1**

Table: /ISV/PAQ0DTX\_UES  
 Displayed Fields: 5 of 5 Fixed Columns: [ 3 ] List Width 0250

MANDT	USEREXIT_NAME	PROJECT	CLASSNAME	ACTIVE	
<input type="checkbox"/>	100	CHANGE_ALV	USEREXITS_DEMOS	ZD260_CL_AE_UE_DEMO	X

Abbildung 18: Definition und Aktivierung der UserExits

### 6.2 AGIMENDO.edit User Exits pflegen

Um die UserExits zu pflegen, findet sich in der Komponente zur Anlage und Verwaltung von Views (siehe Kapitel 3) ein Button „UserExit pflegen“, der ein neues Popup für die Verwaltung der UserExits öffnet (Abbildung 19). Die entsprechende Komponente /ISV/AE\_USER\_EXIT\_CONFIG kann ebenfalls direkt aufgerufen werden.

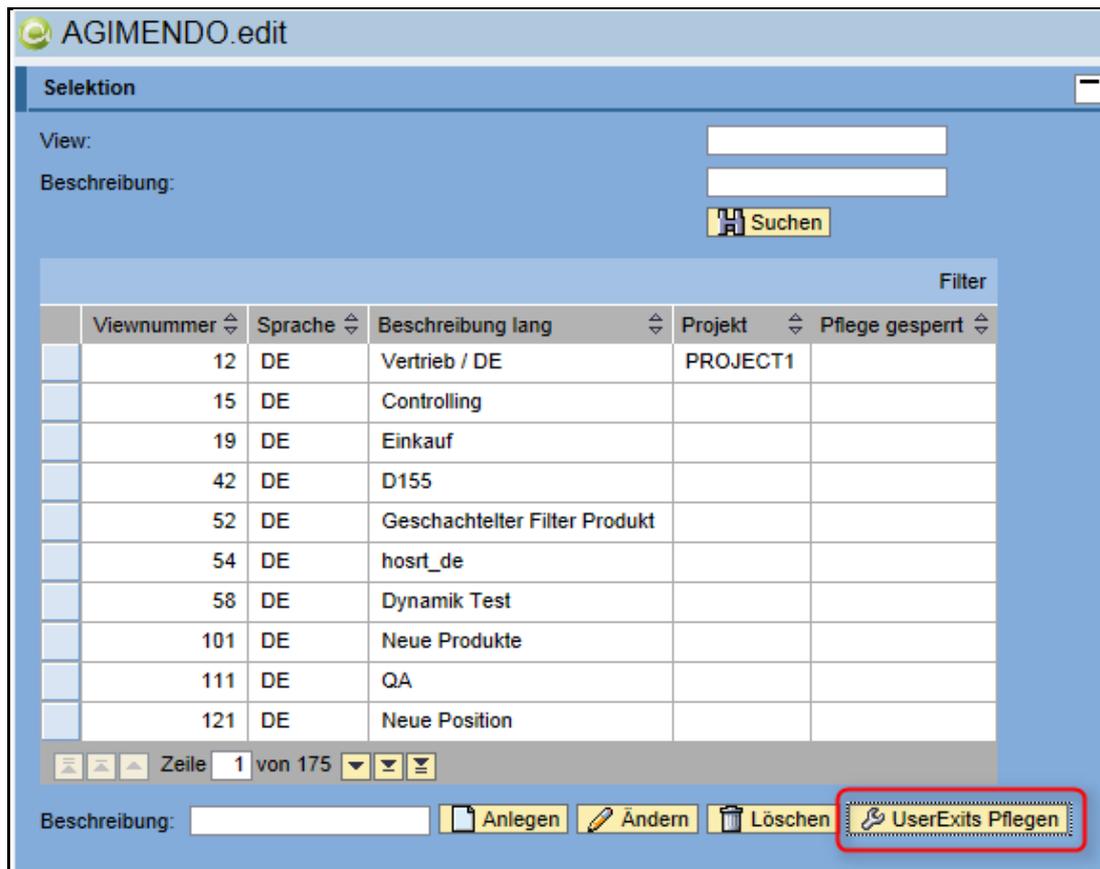


Abbildung 19: "UserExits Pflegen" Button

In diesem Popup sind mehrere Element verfügbar (Abbildung 20):

1. Eine Dropdown Liste mit alle vorhandene Projekte. Die Standarteinstellung ist so dass, das Projekt ohne Name automatisch gewählt wird. Wenn zuvor ein View ausgewählt ist, wird das entsprechende Projekt vorausgewählt.
2. Eine Dropdown Liste mit allen UserExits, die noch nicht dem Projekt zuweisen sind. Der Button „Hinzufügen“ ermöglicht einen ausgewählten UserExit hinzuzufügen. Ein hinzugefügter UserExit ist anfangs inaktiv. Wenn alle UserExits , die schon für das Projekt definiert sind, den gleichen Klassennamen zugeordnet haben, wird dem hinzugefügten UserExit dieser ebenfalls zugeordnet.
3. Eine Tabelle, die alle vorhandenen UserExits für das ausgewählte Projekt zeigt. Nur die „Klassenname“- und „Aktiv“ Felder sind editierbar.

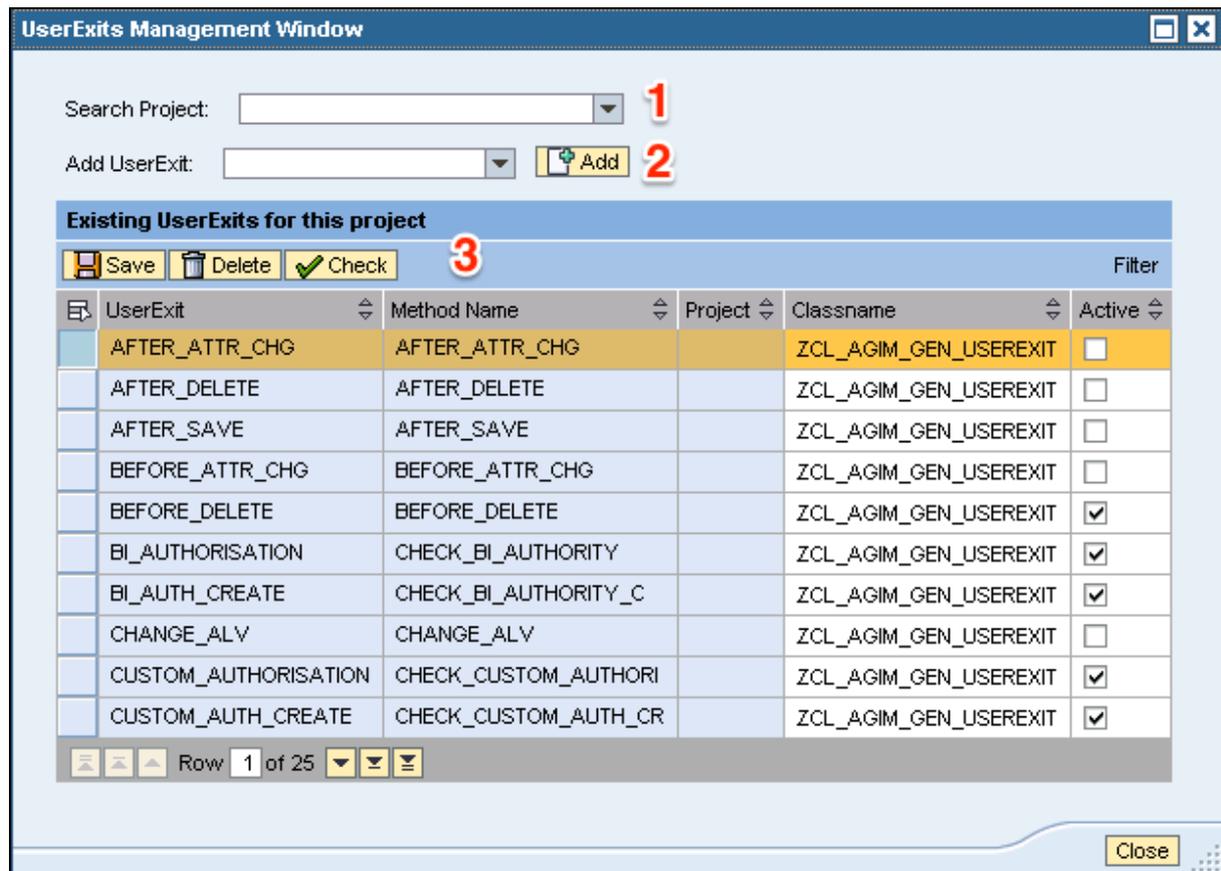


Abbildung 20: Übersicht des Fenster UserExits Verwaltung

In der Tabelle befinden sich drei weitere Buttons: „Sichern“ speichert die Änderungen der Tabelle in der Datenbank (außer dem Hinzufügen / Löschen von Zeilen wird direkt gespeichert). „Löschen“ löscht nach Bestätigung die ausgewählten Zeilen. „Prüfen“ prüft, ob der angegebene Klassennamen gültig ist und die richtige Vererbung aufweist. Wenn dabei ein Fehler gefunden wird, wird eine Fehlermeldung angezeigt, es kann aber trotzdem gespeichert werden.

Zum Anlegen eines neuen UserExits wird zunächst ein Projekt unter "Projekt suchen" ausgewählt. Danach kann unter "UserExit hinzufügen" einer der AGIMENDO.edit UserExit-Funktionen ausgewählt werden, für die eine eigene Funktion definiert werden soll.

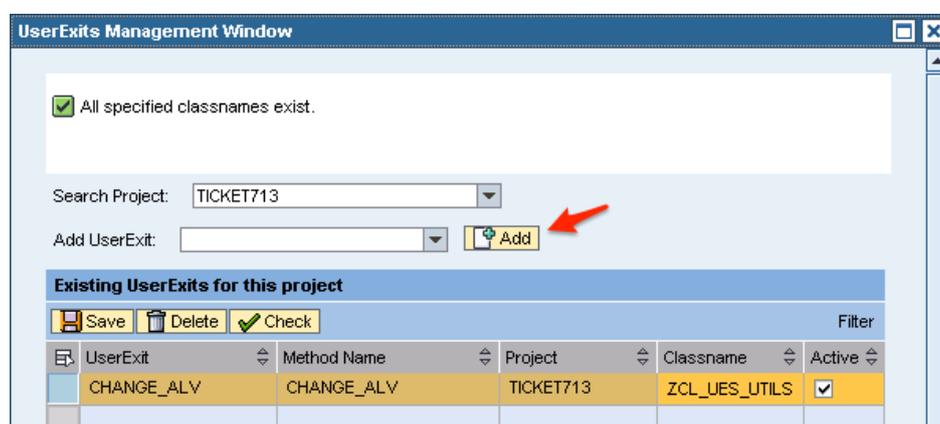


Abbildung 21: Auswahl von Projekt und UserExit

In der Tabelle darunter wird dann der Klassenname angegeben, wobei die Klasse im jeweiligen Projekt existieren muss und eine Redefinition der angegebenen UserExit-Methode beinhalten muss. Dann muss die Regel für diesen UserExit per Checkbox noch auf "aktiv" gesetzt werden.

### 6.3 Übersicht der AGIMENDO.edit User Exits

Die Klasse /ISV/PAQ0DCX\_UE\_IMP beinhaltet Methoden für folgende User Exits:

- AFTER\_ATTR\_CHG -> Aktionen nach dem Start des Attributänderungslaufs
- AFTER\_DELETE -> Aktionen nach dem Löschen eines Datensatzes
- AFTER\_SAVE -> Aktionen nach dem Zwischenspeichern
- BEFORE\_ATTR\_CHG -> Aktionen vor dem Start des Attributänderungslaufs
- BEFORE\_DELETE -> Aktionen vor dem Löschen eines Datensatzes
- BI\_AUTHORISATION -> Überprüfung der Analyseberechtigungen -> Default-Coding
- BI\_AUTH\_CREATE -> Überprüfung der Analyseberechtigungen (Anlage) -> Default-Coding
- CHANGE\_ALV -> Änderung des ALV-Objekts -> Default-Coding
- CUSTOM\_AUTHORISATION -> Überprüfung von „eigenen“ Berechtigungen -> Default-Coding
- CUSTOM\_AUTH\_CREATE -> Überprüfung von „eigenen“ Berechtigungen (Anlage)
- PFCG\_AUTHORISATION -> Überprüfung von PFCG-Rollen bei den Aktionen -> Default-Coding
- CUSTOM\_VALIDATION -> Validierung auf Datensatzebene -> Default-Coding
- FILTER\_LANGU -> Filtern der Sprache
- GET\_VALUE -> Vorbelegung bei der Neuanlage von neuen Datensätzen (bspw. ID-Erzeugung)
- GET\_KEYS\_F\_SELECTION -> Ermittlung des semantischen Schlüssels zur Auswahl (Default Coding)
- CHECK\_PFCG\_AUTHORISATION -> Überprüfung der Berechtigung basiert auf PFCG-Rollen -> (Default Coding)
- CUSTOM\_POPUP\_FIELD -> Überprüfung der Änderungen in den Eingabefeldern in Pop-up-Fenstern (Create/Change/Mass)
- EXECUTE\_BTN\_ACTION -> Ausführung der Aktion für ein kundenspezifisches Button
- EXEC\_BTN\_ACTION\_CLOSED -> Nachbearbeitung einer Custom-UI-Aktion
- CUSTOM\_VALUE\_HELP -> Aufruf von der kundenspezifischen Suchhilfe für ein Attribut
- GET\_NEW\_LINES\_COUNT -> Wiedergabe von der Anzahl von neuen (nicht gespeicherten) Zeilen für einen View
- GET\_SORT\_FULL\_TABLE -> Wiedergabe von den Sortierungskriterien (Liste der Felder) für eine Datentabelle
- GET\_TIMEDEP\_LANGDEP\_TEXT Liefert den Text für ein nachgelesenes Attribut und kann hierbei Einfluss auf Zeitabhängigkeit / Sprachabhängigkeit nehmen
- READ\_ONLY\_MODE -> Prüfen ob nur gelesen werden darf oder auch geändert (überschreibt Wizard-Einstellungen)
- SKIP\_ATTRIB\_CHANGE -> Prüfung, ob der Attributsänderungslauf durchgeführt werden soll (überschreibt Wizard-Einstellungen)
- SHOW\_BUTTONS -> Wiedergabe von der Liste der Buttons für den aktuellen View
- TEST\_FOR\_MANDATORY\_FIELD\_CHECK -> Überprüfung ob die obligatorischen Felder gefüllt sind
- TEXT\_CHANGED -> Text eines Attributs wurde geändert

**Hinweis:** Im Anhang 10.2 Referenz der UserExit ist eine Referenz aller UserExits mit Parametern / Rückgabewerten etc. zu finden.

**Hinweis:** In meisten Fällen entspricht der Name des User Exits dem Namen der Methode aus /ISV/PAQ0DCX\_UE\_ABSTRACT. Die Abweichungen zwischen den Namen des User Exits und der entsprechenden Methode sind in der folgenden Übersicht aufgelistet:

User Exit (Konfiguration in DB Tabelle)	Methode
BI_AUTHORISATION	CHECK_BI_AUTHORITY
CUSTOM_AUTHORISATION	CHECK_CUSTOM_AUTHORITY
BI_AUTH_CREATE	CHECK_BI_AUTHORITY_CREATE
CUSTOM_AUTH_CREATE	CHECK_CUSTOM_AUTH_CREATE
CUSTOM_VALIDATION	CHECK_VALIDATION
PFCG_AUTHORISATION	CHECK_PFCG_AUTHORITY
EXECUTE_BTN_ACTION	EXECUTE_BUTTON_ACTION
CUSTOM_VALUE_HELP	GET_CUSTOM_VALUE_HELP
SHOW_BUTTONS	GET_UE_BUTTONS_FOR_LOCATION
EXEC_BTN_ACTION_CLOS	EXEC_BTN_ACTION_CLOSED

**Hinweis:** Unter „Default-Coding“ ist zu verstehen, dass die Funktionalität durch das ausgelieferte Coding in der Klasse /ISV/PAQ0DCX\_UE\_IMP realisiert wird, bei Bedarf kann es jedoch manuell erweitert bzw. ersetzt werden.

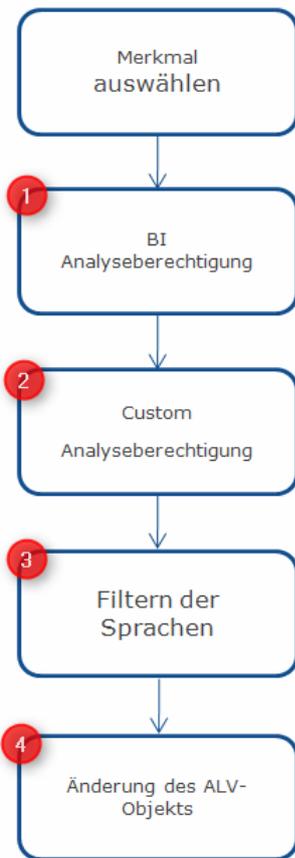
**Hinweis:** Im UserExit CUSTOM\_VALIDATION erfolgt im Default-Coding die Prüfung auf gültige InfoObjekt-Werte. Hier wird überprüft ob der ausgewählte Wert im BasisinfoObjekt des Attributs in den Stammdaten existiert. Diese Default-Überprüfung schlägt bei verschiedenen Standard-InfoObjekten (z.B. OFISCPER oder OCALMONTH) und bei eigenen InfoObjekten ohne Stammdaten fehl. Aus diesem Grund müssen im Falle einer Nutzung der Standard-Werteüberprüfung bei InfoObjekten ohne Stammdaten die Überprüfung manuell ausgeschlossen werden. Zu diesem Zweck existiert im Default-Coding eine Methode mit dem Namen TEST\_FOR\_EXCLUDE\_FROM\_CHECK, die redefiniert werden muss und in deren Coding für jedes Merkmal geprüft wird, ob eine Überprüfung stattfinden soll. Im Kapitel 10.1.1 wird beispielhaft dargestellt, wie dieses zu implementieren ist.

## 6.4 AGIMENDO.edit Ausführungsreihenfolge User Exits

### 6.4.1 User Exits bei der Auswahl des zu bearbeitenden Merkmals

Bei der Pflege von Datensätzen mit Hilfe von AGIMENDO.edit kann die angezeigte Datenmenge (Stammdaten eines Merkmals) eingeschränkt werden. Die als Berechtigung zu definierenden Kriterien lassen sich nicht nur aus den SAP BW Analyseberechtigungen ableiten, da auch Merkmale eingeschränkt werden sollen, die im SAP BW Standard als „nicht berechtigungsrelevant“ definiert wurden. Hintergrund sind zu vermeidende Seiteneffekte mit anderen (Reporting-)Anwendungen.

Daten werden aus den SAP BW Standardtabellen in eine interne Tabelle gelesen. Die interne Tabelle wird dem User Exits übergeben. Innerhalb des User Exits kann die Datenmenge manipuliert werden. Die Darstellung der ggf. veränderten Datenmenge kann durch einen weiteren User Exit verändert werden. Schließlich werden die Daten in der Pflegeoberfläche der Stammdatenpflege zur Anzeige gebracht.



**(1) BI Analyseberechtigung (BI\_AUTHORISATION):** Falls das Merkmal als „berechtigungsrelevant“ definiert wurde, wird die angezeigte Datenmenge (Stammdaten dieses Merkmals) anhand den Einschränkungskriterien aus BI eingeschränkt.

**(2) Custom Analyseberechtigung (CUSTOM\_AUTHORISATION):** Falls Einschränkungen für die Attribute dieses Merkmals in der entsprechende Customizing Tabelle definiert wurden, wird die angezeigte Datenmenge (Stammdaten dieses Merkmals) anhand der Einschränkungskriterien aus der Customizing Tabelle eingeschränkt.

**(3) Filtern der Sprachen (FILTER\_LANGU):** Anhand dieses User Exits kann bestimmt werden, dass die anzuzeigende Datenmenge auf die Sätze bestimmter Sprachen begrenzt wird.

**(4) Änderung des ALV-Objekts (CHANGE\_ALV):** Anhand dieses User Exits kann die Darstellung der anzuzeigenden Datenmenge beeinflusst werden.

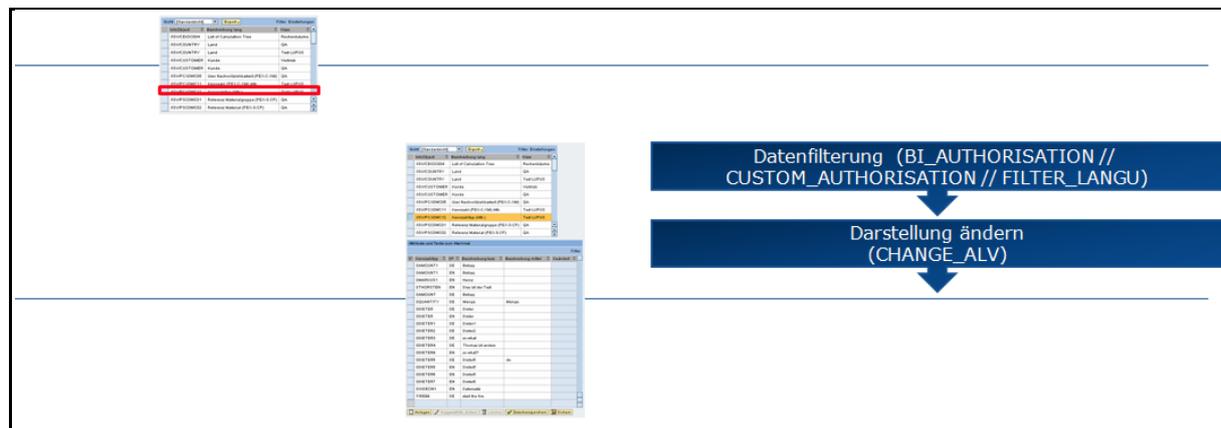


Abbildung 23: User Exit bei der Auswahl des zu bearbeitenden Merkmals

### 6.4.2 User Exits beim Anlegen von Datensätzen

Beim Anlegen eines neuen Datensatzes wird entsprechend den vorgenommenen Einstellungen im Wizard (UE-Anlage aus Kapitel 4.3.6) das betreffende Feld bereits durch den Aufruf des „GET\_VALUE“ User Exit vorbelegt.

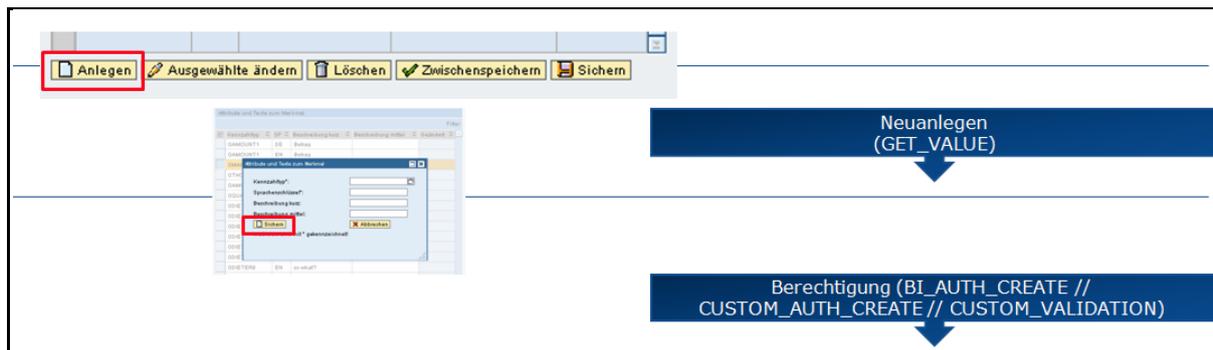


Abbildung24: User Exit beim Anlegen von Datensätzen

### 6.4.3 User Exits beim Speichern von Datensätzen

Durch Aufruf der folgenden User Exits können unter anderem neu angelegte bzw. geänderte Datensätze gegen bestehende Stammdaten geprüft werden (CUSTOM\_VALIDATION). Datensätze dürfen dann nur gespeichert werden, wenn die Eingaben für ausgewählte Felder (Attribute) bereits als Stammdaten der jeweiligen Merkmale vorhanden sind.

Innerhalb des User Exits kann die Entscheidung erfolgen, ob das Speichern zulässig ist oder nicht. Bei fehlgeschlagener Überprüfung wird das Speichern verhindert und eine Textnachricht (Kurztext) wird erzeugt. Die Nachricht wird in dem Popup angezeigt, in dem die Daten gepflegt werden.

Ergibt die Überprüfung keine Fehler, werden die Änderungen zwischengespeichert. Während des User Exits „AFTER\_SAVE“ wird ein Attributänderungslauf durchlaufen, der die erzeugten oder geänderten Attribute im System aktualisiert (BEFORE\_ATTR\_CHG/ AFTER\_ATTR\_CHG). Danach werden die vier User Exits ausgeführt, die in Kapitel 6.4.3 bereits beschrieben wurden.

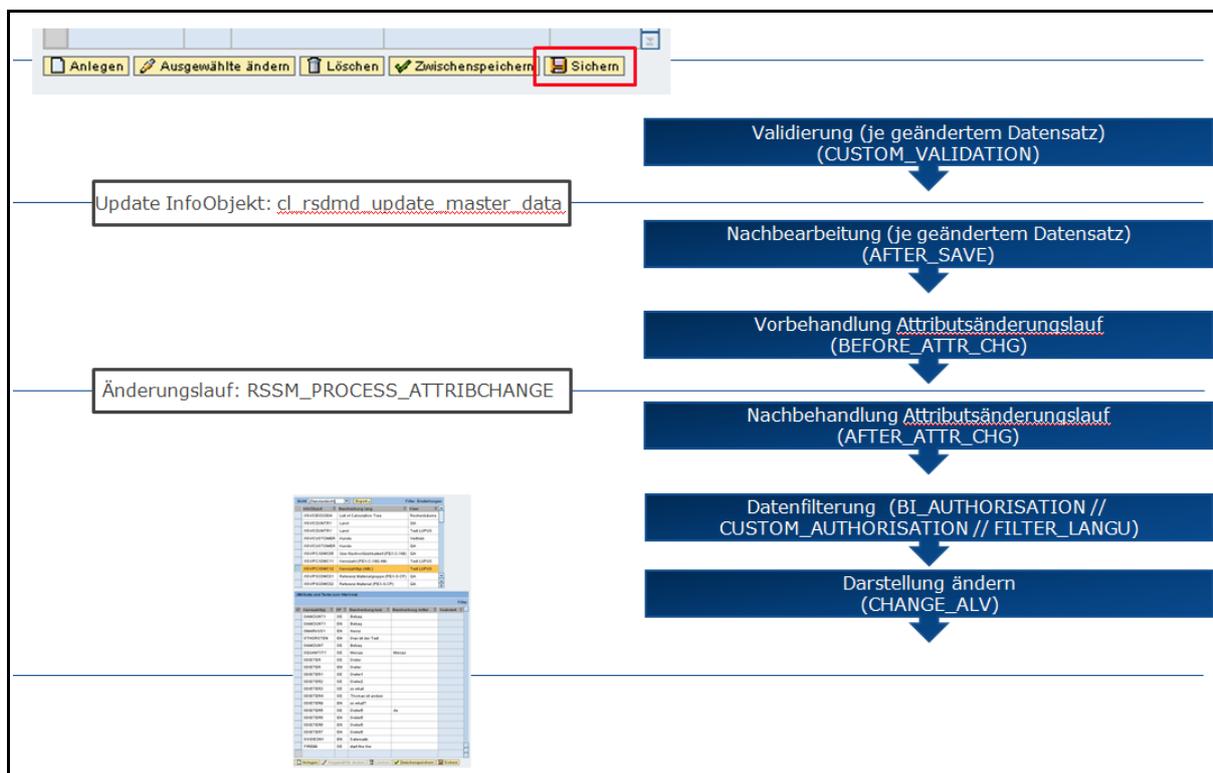


Abbildung 25: User Exit beim Speichern von Datensätzen

Werden Datensätze direkt im Feld geändert oder über den Button „Ausgewählte ändern“ bearbeitet, werden die Änderungen nicht in der Datenbank gespeichert, sondern nur in der Tabelle übernommen. Das Speichern und damit das Schreiben in die Datenbank müssen über den Button „Speichern“ geschehen.



Abbildung 26: Kein User Exit bei "Änderungen übernehmen"

#### 6.4.4 User Exits beim Zwischenspeichern von Datensätzen

Beim Zwischenspeichern von geänderten Datensätzen wird als erstes der User-Exit „CUSTOM\_VALIDATION“ aufgerufen. Danach wird der User Exit „AFTER\_SAVE“ aufgerufen, der die Änderungen in der Datenbank aktualisiert.

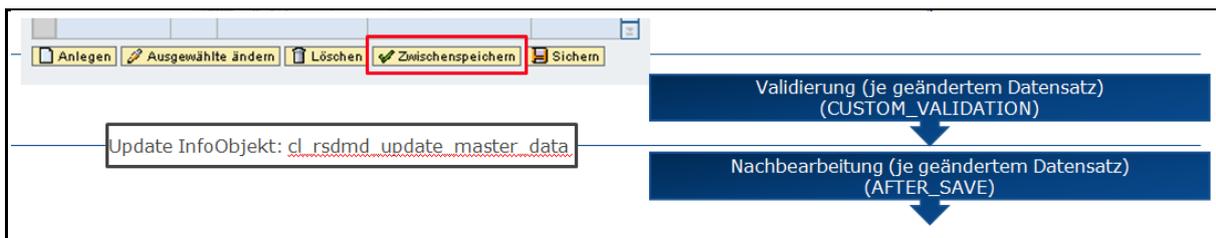


Abbildung 27: User Exit beim Zwischenspeichern von Datensätzen

#### 6.4.5 User Exits beim Löschen von Datensätzen

Beim Löschen von Datensätzen wird zuerst eine Löschabfrage gestellt („BEFORE\_DELETE“), bevor der User Exit „AFTER\_DELETE“ aufgerufen wird, der die ausgewählten Stammdaten aus der Datenbank löscht und diese aktualisiert.

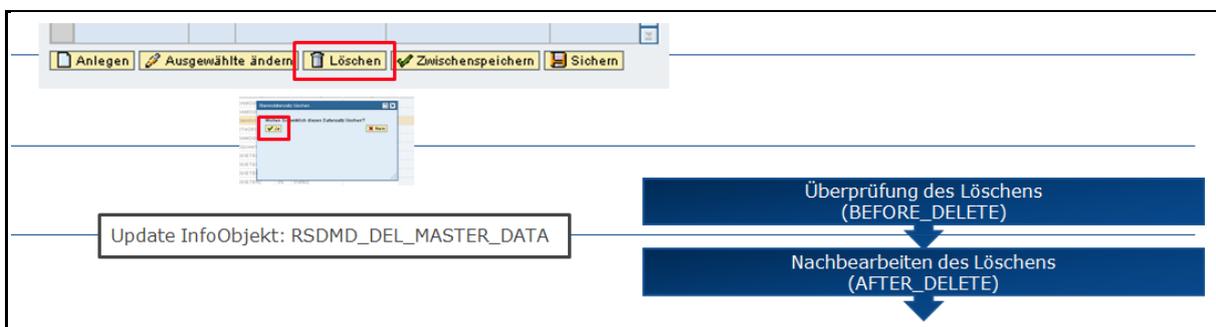


Abbildung 28: User Exit beim Löschen von Datensätzen

## 7. Berechtigungen

Über das Berechtigungsobjekt /ISV/AEDIT kann der Zugriff auf die Pflegeoberfläche gesteuert werden. In der Abbildung29 sind die zur Verfügung stehenden und berücksichtigten Aktivitäten dargestellt.

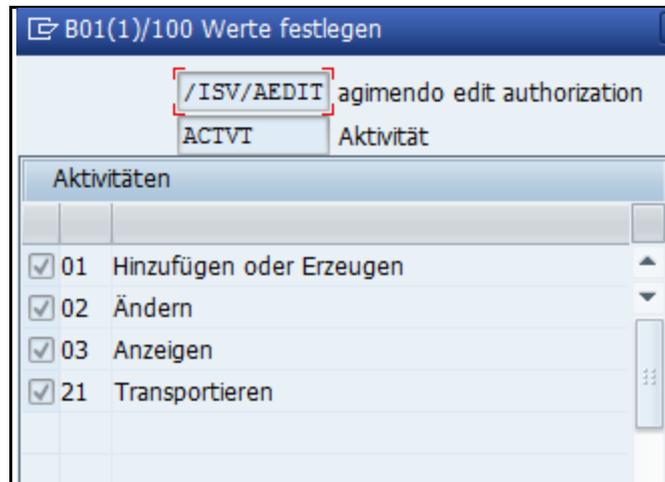


Abbildung29: Aktivitäten des Berechtigungsobjekts /ISV/AEDIT

Im Code der AGIMENDO.edit WebDynpro Komponenten wird es durch die Anweisung AUTHORITY-CHECK geprüft, ob ein Benutzer eine ausreichende Berechtigung für eine bestimmte Aktivität hat. Dazu wird im Stammsatz des Benutzers in den dort angegebenen Profilen nach Berechtigungen für das angegebene Berechtigungsobjekt (/ISV/AEDIT) gesucht. Wenn eine der gefundenen Berechtigungen mit den vorgegebenen Werten übereinstimmt, dann war die Prüfung erfolgreich.

Die Aktivität 01 (Hinzufügen oder Erzeugen) wird beim Öffnen der Anlage- und Berechtigungskomponente von AGIMENDO.edit geprüft. Die Aktivität 02 (Ändern) wird beim Öffnen der Wizard-Komponente zum Ändern eines Views geprüft. Die Aktivität 03 (Anzeigen) wird beim Öffnen der tatsächlichen Stammdatenpflegeanwendung geprüft.

Die Aktivität 21 wird beim Öffnen des Transportmanagers geprüft bzw. wird geprüft um zu entscheiden ob die Optionen zum Transport im Wizard eingeblendet werden oder nicht.

## 8. Customizing

AGIMENDO.edit kann ab Version 2.12.8 an spezielle Systemlandschaften und Anforderungen angepasst werden. Im Normalfall müssen die hier beschriebenen Parameter nicht angepasst werden. Alle nachfolgenden Einstellungen werden in der Tabelle /ISV/AE\_SETTING vorgenommen.

Name	Value	Beschreibung
ATTRIB_CHANGE_WAIT_STRATEGY	INCR   STAT   RAND	Dieser Parameter beschreibt die Änderung der Wartestrategie wenn bereits ein Attributänderungslauf aktiv ist. INCR ist standardmäßig eingestellt und wartet bis max. 10x

jeweils 2, 3, ..., 10 Sekunden lang.  
STAT wartet immer 3 Sekunden und  
RAND zwischen 3 und 8 Sekunden.

## 9. Transportwesen

Mit der Produktversion 2.12 wurde ein neuer Transportanschluss eingeführt. Damit verfügt AGIMENDO.edit über die Möglichkeit, Projekte-, Views-, InfoObjekt-View Zuordnungen separat zu transportieren. Der bisher empfohlene Weg, alle Tabellen in Summe zu transportieren ist damit nicht mehr der präferierte.

### 9.1 Überblick

Das Transportwesen von AGIMENDO.edit nutzt für den Transfer der Informationen zwischen den Systemen das Standardtransportwesen der SAP. Für einen Transport wird daher ein offener Workbench-Auftrag benötigt, der dem aktuellen Benutzer zugeordnet ist. Auf diesen Auftrag werden die AGIMENDO.edit Inhalte in sog. Transport-Items geschrieben. Ein Transport-Item kann entweder ein Projekt (vollumfänglich mit allen Views und allen darin enthaltenen InfoObjekten), einen View (also alle InfoObjekte des Views) oder nur ein einzelnes InfoObjekt im Kontext eines View enthalten. Beim Erstellen eines Transportitems wird die View-Definition kopiert. Eine nachträgliche Änderung der View-Definition ändert den Inhalt des Transportitems nicht mehr, auch wenn der zu Grunde liegende SAP-Transportauftrag noch nicht freigegeben wurde. Nach einem Transport des zugrundeliegenden Auftrags in das Zielsystem können dann über den AGIMENDO.edit Transportmanager die (auf SAP Transportebene) eingespielten Transport-Items ausgepackt und importiert werden. Auf diesem Wege können auch frühere Transporte einfach über die WebDynpro-Oberflächen des Transportmanagers erneut importiert werden.

Es ist möglich ein Transportitem als „Manuellen Transport“ zu markieren. Dies bedeutet technisch, dass der Tabelleneintrag nicht auf einen SAP-Transportauftrag geschrieben wird. Dieses Transportitem kann dann entweder manuell auf einen Transportauftrag geschrieben werden oder nachträglich mit Hilfe des Transportmanagers mit einem bestehenden Transportauftrag verknüpft werden (siehe dazu Kapitel 8.3.1)

**Hinweis:** Wenn in diesem Kapitel von dem Transport von InfoObjekten gesprochen wird, dann ist hiermit nicht der Transport des InfoObjekts an sich oder dessen Inhalt sondern lediglich dessen Konfiguration in AGIMENDO.edit im Kontext eines Views gemeint.

#### 9.1.1 Trennung der Systeme

AGIMENDO.edit vergibt beim Anlegen eines Views eine neue View-Nummer. Diese View-Nummer wird über einen Nummernkreis ermittelt. Damit beim Transport zwischen den Systemen von einzelnen Views keine Konflikte der Views-Nummern auftreten, wird empfohlen die Nummernkreise in den verschiedenen Systemen unterschiedliche einzustellen (bspw. 0\* für das Entwicklungssystem, 1\* für das Qualitätssicherungssystem und 2\* für das Produktionssystem)

Bei der Zuordnung von InfoObjekten in einen View wird für diese Zuordnung ein Quellsystem definiert. Bei jeder Änderung der Definition wird eine Version (letzter Änderer etc.) geschrieben. Auf diesem Weg kann dann ein Transportstatus ermittelt und angezeigt werden. („Ist das Objekt seit dem letzten Transport geändert worden?“)

**Hinweis:** AGIMENDO.edit zählt zwar eine Versionsnummer hoch, das Datenmodell ist aber nicht streng versioniert. Eine Rückkehr zu einem alten Stand ist nur durch das Wiedereinspielen eines Transportitems möglich. (Siehe dazu Kapitel 8.2.2)

## 9.2 Transportieren über den Wizard

Aus der Oberfläche des Wizards heraus kann auf zwei Ebenen ein neues Transportitem angelegt werden.

### 9.2.1 Transport eines vollständigen Views

Im Ersten Schritt des Wizard kann ein ausgewählter View über das Transportsymbol (  Laster) transportiert werden. Im folgenden Popup können neben den allgemeinen Einstellungen (zugrundeliegender SAP Transportauftrag und Dokumentationstext) auch definiert werden, ob die View Berechtigungen mit transportiert werden sollen oder nicht. Die Option des „Manual Transport“ bedeutet, dass das Transport Item zwar erzeugt wird, aber nicht einem SAP Transportauftrag hinzugefügt wird. Das Transport-Item wird in der Tabelle /ISV/TRANSIT gespeichert und kann von dort manuell transportiert werden oder zu einem späteren Zeitpunkt einem SAP Transportauftrag zugeordnet werden.

Oft werden im Produktionssystem andere Berechtigungen vergeben als in Entwicklung oder Test. Daher werden im Standardfall die Berechtigungen nicht mittransportiert.

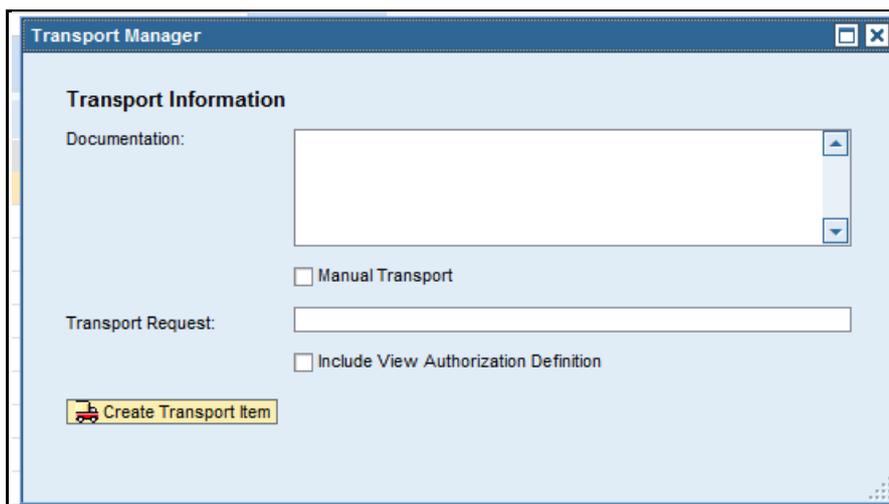


Abbildung30: Optionen beim Transport eines Views

### 9.2.2 Transport eines einzelnen InfoObjekts in einem View

Im zweiten Schritt des Wizard können die Kombinationen aus InfoObjekt und View transportiert werden. Dies geschieht durch die Auswahl eines zugeordneten InfoObjekts und dann durch einen Klick auf das Transportsymbol (  Laster). Dadurch öffnet sich ein Popup in dem zusätzliche

Informationen zum Transportitem angegeben werden müssen. Hier kann nur der optionale manuelle Transport gewählt werde, Berechtigungen können nur mit dem vollständigen View transportiert werden (siehe oben).

In der Abbildung 27 ist der zweite Schritt des Wizard dargestellt. Die drei Icons haben dabei die folgende Semantik:

-  Grau bedeutet, dieses InfoObjekt wurde in diesem View noch nicht transportiert.
-  Gelb bedeutet, dieses InfoObjekt wurde in diesem View bereits transportiert wurde aber danach nochmal geändert. Im Zielsystem hat gelb zusätzlich die Semantik, dass hier wohl eine Änderung im Zielsystem vorliegt, die durch einen Import einer neuen Version verloren geht.
-  Grün bedeutet, dieses InfoObjekt wurde in diesem View transportiert und hat seit dem keine Änderung erfahren.



Abbildung31: Wizard - Anzeige von Transport-Status

Im dritten Schritt des Wizard können im Reiter „Technische Einstellungen“ weitere Einstellungen zur Währungsdarstellung und zum Tabellenverhalten sowie zusätzliche Informationen zum Transportstatus eingesehen werden. Folgende Einstellungen sind wählbar:

- Währungsdarstellung - stellt Währungen, die intern mit verschobenen Dezimalstellen gespeichert werden, mit entsprechend mehr Dezimalstellen dar
- Tabellenverhalten - merkt sich Scrollposition und gesetzte Filter von Tabellen beim Neuanlegen und Erstellen von Tabelleneinträgen sowie beim Sichern (Die Daten werden dabei trotzdem neu geladen). Wenn diese Option nicht ausgewählt ist, wird die Datentabelle bei jedem Neuladen in die Ausgangsposition zurückgesetzt.

Zum Transportstatus werden folgende Informationen angezeigt:

- Version – bei jeder Änderung erhöht sich die Version um eins,
- Quellsystem – die SID des Systems, in dem die InfoObjekt/View Kombination angelegt wurde
- Letzer Änderer – Benutzername, Datum und Uhrzeit der letzten Änderung
- Transportitem – Nummer des Transport-Items, dem das Objekt zuletzt zugeordnet wurde, leer wenn bisher noch nie einem Transport-Item zugeordnet, die Transportitem-Nummer beginnt immer mit der SID des Quellsystems (bspw. B01
- Transportstatus – Wenn das Objekt seit der letzten Zuordnung geändert wurde

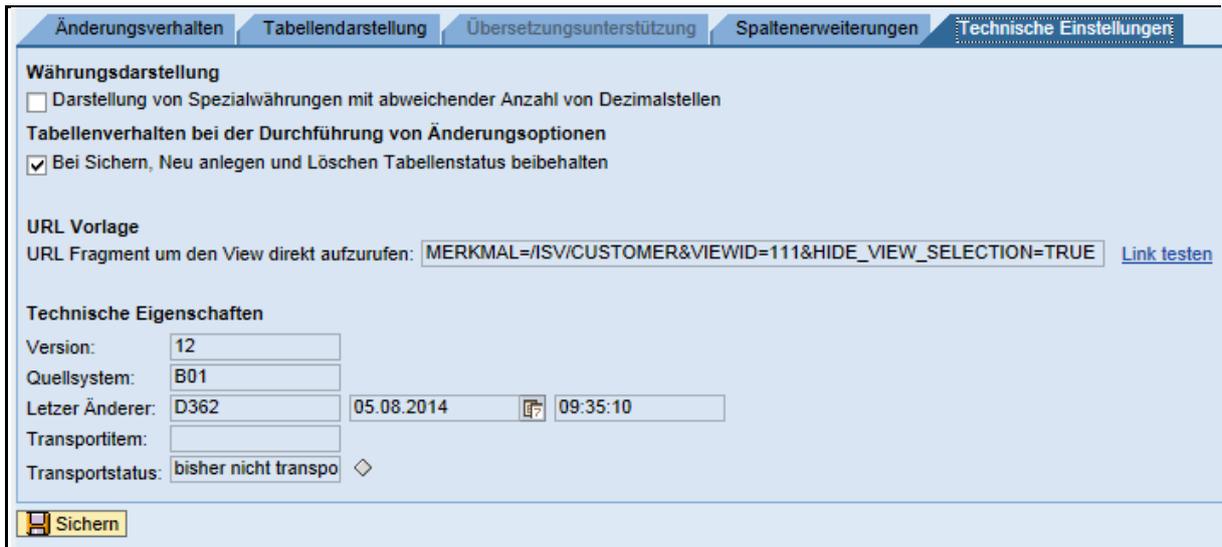


Abbildung32: Wizard - Technische Einstellungen zum Transportwesen

Wenn im Wizard ein InfoObjekt bearbeitet wird, das nicht im aktuellen System angelegt wurde, wird eine Wartung ausgegeben. Im Beispiel wurde der View im System mit der SID C01 angelegt und wird in einem System mit der SID B01 gerade bearbeitet. Eine Änderung würde bei einem erneuten Transport von dem System C01 nach B01 verloren gehen.

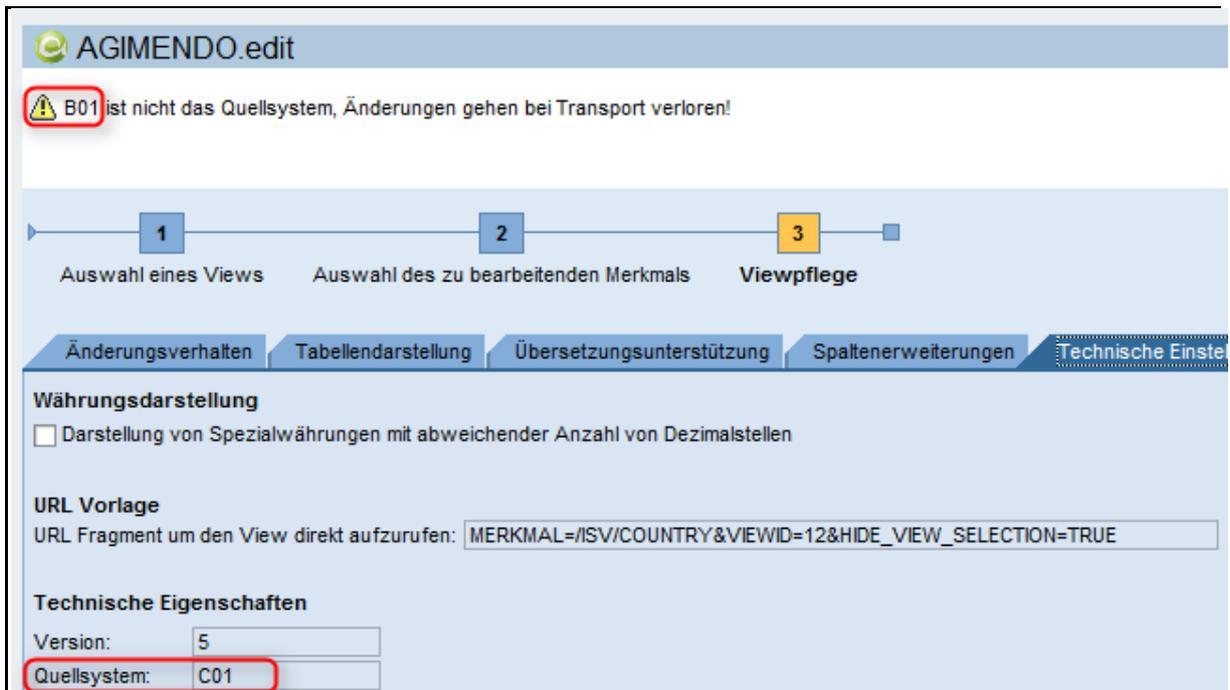


Abbildung33: Wizardwarnung bei Bearbeitung im Nicht-Quellsystem

### 9.3 Transportmanager

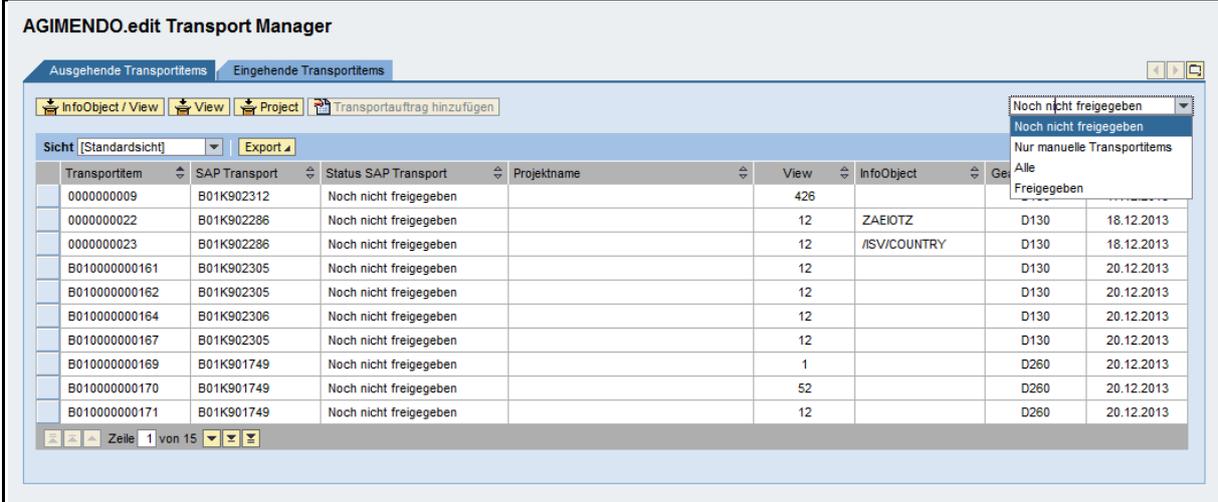
Der Transportmanager besteht als separate WebDynpro-Anwendung die auch separat berechtigt werden kann (siehe dazu Kapitel 7). Der Aufruf des Transportmanagers erfolgt über folgende URL

[http://<server>:<port>/sap/bc/webdynpro/isv/ae\\_transport\\_manager](http://<server>:<port>/sap/bc/webdynpro/isv/ae_transport_manager)

Die Oberflächen des Transportmanagers gliedern sich in die zwei Bereiche „Ausgehende Transportitems“ und „Eingehende Transportitems“.

### 9.3.1 Ausgehende Transportitems

Im Bereich der Ausgehenden Transportitems können die vorhandenen Transportitems eingesehen werden und neue Transportitems angelegt werden. In der folgenden Abbildung ist die Sicht der Ausgehenden Transporte dargestellt.



Transportitem	SAP Transport	Status SAP Transport	Projektname	View	InfoObject	Ge	
0000000009	B01K902312	Noch nicht freigegeben		426			
0000000022	B01K902286	Noch nicht freigegeben		12	ZAEIOTZ	D130	18.12.2013
0000000023	B01K902286	Noch nicht freigegeben		12	/ISV/COUNTRY	D130	18.12.2013
B01000000161	B01K902305	Noch nicht freigegeben		12		D130	20.12.2013
B01000000162	B01K902305	Noch nicht freigegeben		12		D130	20.12.2013
B01000000164	B01K902306	Noch nicht freigegeben		12		D130	20.12.2013
B01000000167	B01K902305	Noch nicht freigegeben		12		D130	20.12.2013
B01000000169	B01K901749	Noch nicht freigegeben		1		D260	20.12.2013
B01000000170	B01K901749	Noch nicht freigegeben		52		D260	20.12.2013
B01000000171	B01K901749	Noch nicht freigegeben		12		D260	20.12.2013

Abbildung34: Ausgehende Transporte im AGIMENDO.edit Transportmanager

In der Liste der Transportitems werden nur Transportitems aufgelistet, die in im aktuellen System erstellt wurden. Über die Auswahlbox oben rechts kann zwischen vier Sichten umgeschaltet werden.

- **Noch nicht freigegeben** – Listet alle Transportitems auf, die einem SAP Transportauftrag zugeordnet sind, der noch nicht freigegeben ist.
- **Nur manuelle Transportitems** – Listet alle Transportitems auf, die noch keinem SAP Transportauftrag zugeordnet sind.
- **Alle** – Listet alle vorhandenen Transportitems auf
- **Freigegeben** – Listet alle Transportitems auf, die einem SAP Transportauftrag zugeordnet sind, der bereitfreigegeben wurde.

Um ein neues Transportitem im Transportmanager anzulegen gibt es drei Möglichkeiten. Es kann ein Transportitem angelegt werden,

- dass ein einzelnes InfoObjekt innerhalb eines Views enthält (  )
- dass einen vollständigen View enthält, also alle zugeordneten InfoObjekte (  )
- dass ein vollständiges Projekt enthält, also alle zugeordneten vollständigen Views (  )

In der folgenden Abbildung 31 ist das Fenster dargestellt, das zur Erstellung von Transportitems auf der untersten Ebene des InfoObjekts verwendet wird. Der angezeigte Status entspricht dem Transportstatus im Wizard. (siehe Kapitel 8.2.2). Es kann ein InfoObjekt im Kontext eines Views und eines Projekts ausgewählt werden. Die Dokumentation dient zur Information und kann als Freitext genutzt werden. Dieser Text wird im Transportmanager des Zielsystems wieder angezeigt.

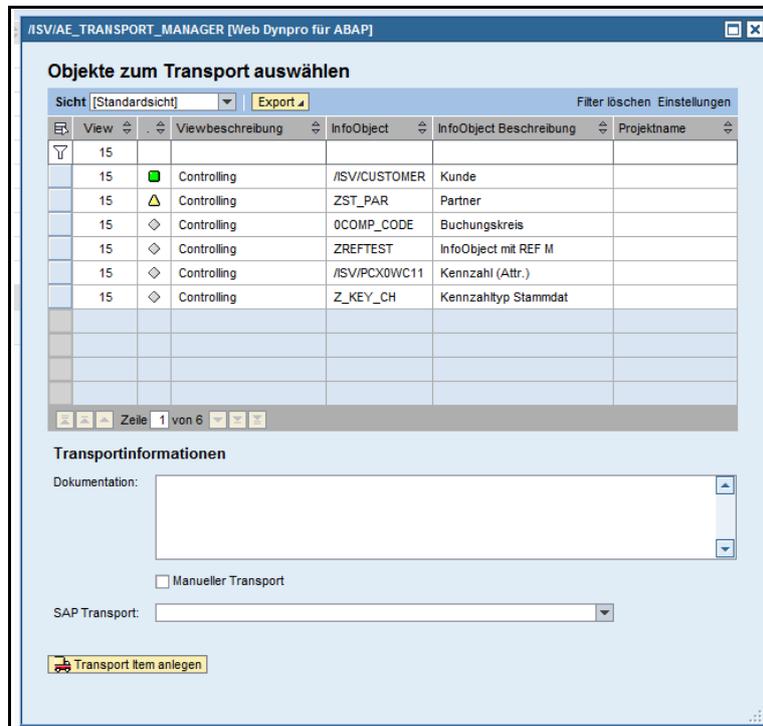


Abbildung35: Transportmanager - Einstellungen zum Transportieren eines InfoObjekt/View

In der folgenden Abbildung 32 ist das Fenster zur Auswahl des zu transportierenden View zu sehen. In der Spalte Anzahl ist die Anzahl der zugeordneten InfoObjekte des Views zu sehen.

Beim Transportieren kann neben der Auswahl des Views und der Angabe des Transportauftrags zusätzlich definiert werden, ob die Berechtigungsinformationen zum View mittransportiert werden sollen oder nicht. (Zugeordnete Benutzer und Rollen).

In jedem Fall werden die Texte zu einem View mit in das Transport-Item gepackt.

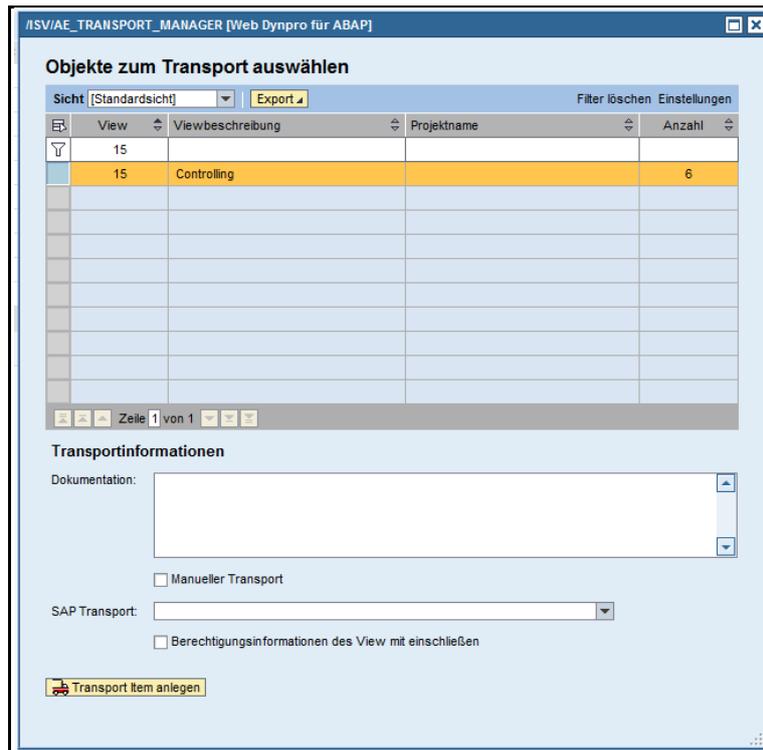


Abbildung36: Transportmanager - Einstellungen zum Transport eines Views

In der Abbildung 33 ist das Fenster dargestellt, das zum Transport eines vollständigen Projekts genutzt wird. Ein Projekt beinhaltet alle Views die dem Projekt zugeordnet sind. Dabei kann ausgewählt werden, ob für alle eingeschlossenen Views neben den InfoObjekten auch die Berechtigungsinformationen transportiert werden sollen. Als zusätzliche weitere Option kann ausgewählt werden, ob die UserExit Konfigurationen des Projekts mittransportiert werden sollen. (Die UserExit Definitionen erfolgen ja im Kontext eines Projekts, siehe Kapitel 6.1).

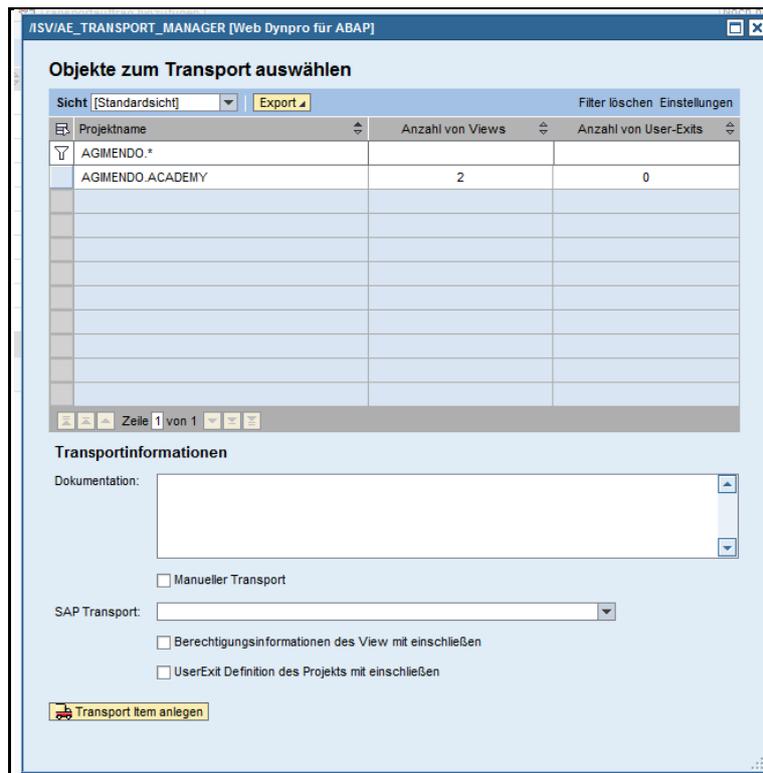


Abbildung37: Transportmanager - Einstellung zum Transport eines Projekts

Wenn für ein Transportitem beim Erstellen als „Manueller Transport“ markiert wurde, kann es nachträglich im Transport Manager einem SAP Transportauftrag zugewiesen werden. Dieser Vorgang ist in der Abbildung 34 dargestellt.

Während der Transportauftrag zugewiesen wird, kann die Dokumentation des Transportitems angepasst werden, bspw. kann eine Notiz eingefügt werden, warum dieser Auftrag gewählt wurde.

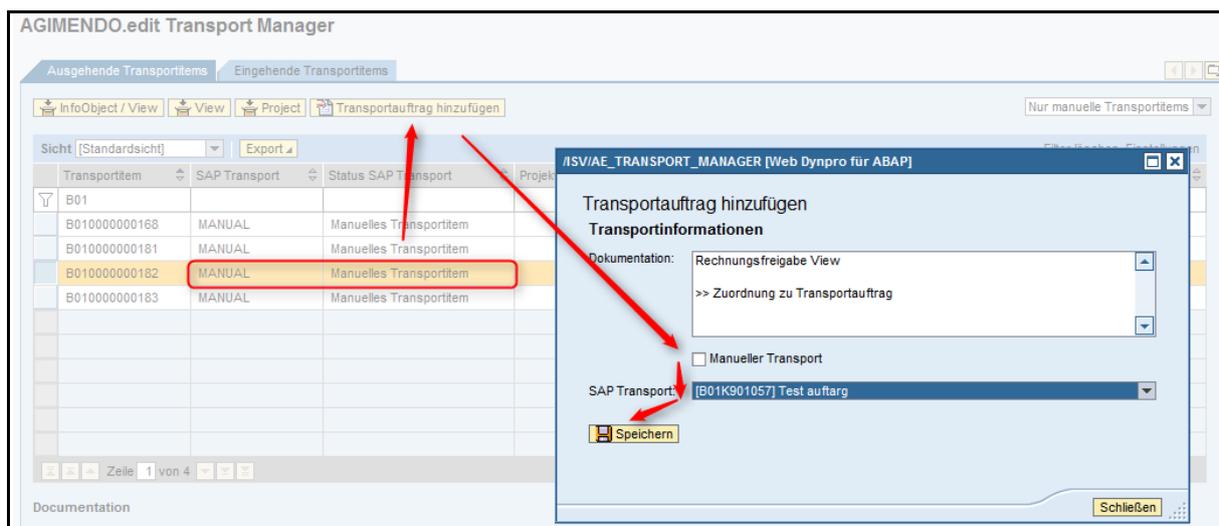


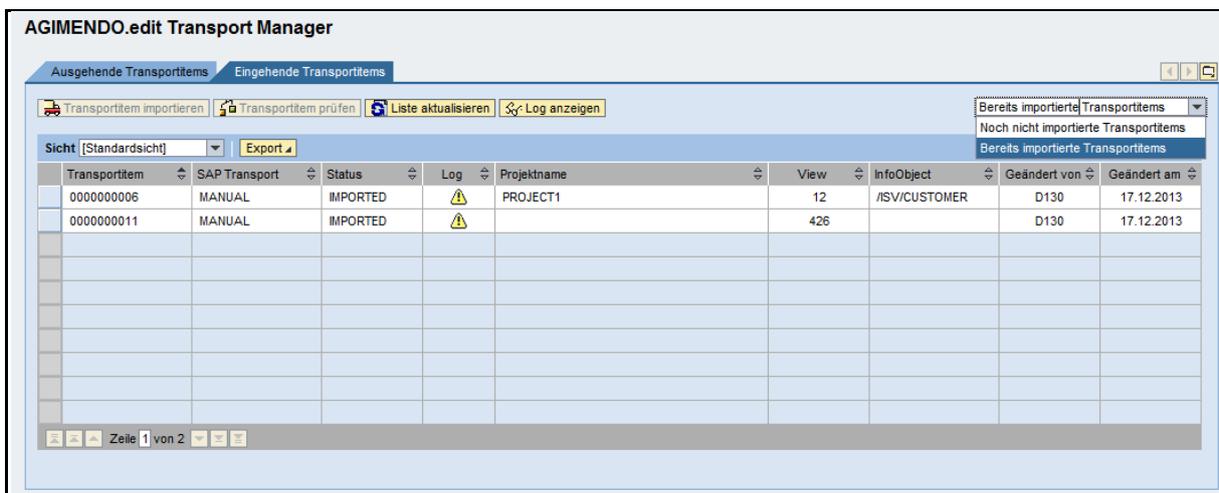
Abbildung38: Transportmanager - Manueller Transport: Zuweisung eines Transportauftrags

### 9.3.2 Eingehende Transportitems

Auf der zweiten Registerkarte können die eingehenden Transportitems eingesehen werden. Dabei kann die Liste gefiltert werden auf entweder bereits importierten Transportitems oder alle noch nicht importierten Transportitems.

Ein ausgewähltes Transportitem kann entweder geprüft (  **Transportitem prüfen** ) oder geprüft und importiert (  **Transportitem importieren** ) werden. Zu einem bereits importierten Transportitem kann das Prüflog des Imports (  **Log anzeigen** ) eingesehen werden.

In der Tabelle wird im Feld „Log“ ein Icon angezeigt, das die schwerwiegendste Meldung des letzten Imports anzeigt.



Transportitem	SAP Transport	Status	Log	Projektname	View	InfoObject	Geändert von	Geändert am
0000000006	MANUAL	IMPORTED		PROJECT1	12	/ISV/CUSTOMER	D130	17.12.2013
0000000011	MANUAL	IMPORTED			426		D130	17.12.2013

Abbildung39: Transportmanager - Eingehende Transporte

### 9.3.3 Transportwesen als Backup

Mit einem kleinen Zusatzprogramm im Backend kann die Transportfunktionalität von AGIMENDO.edit dazu verwendet werden, über ein Transportitem von gepflegten Objekten ein Backup anzulegen und wieder einzuspielen. Dazu muss man zunächst ein Transportitem erstellen (mit dem Inhalt der gesichert und im selben System wieder importiert werden soll) und anschließend die Quellsystem-ID ändern damit das Transportitem in der Liste der Eingehenden Transportitems angezeigt wird. Dazu gibt es das Programm /ISV/AE\_TM\_PREPARE\_REIMPORT. Dieses kann über die SE38 im Backend ausgeführt werden.

In Abbildung40 und in Abbildung41 ist dargestellt wie das Programm in der SE38 aufgerufen und ausgeführt werden kann. Beim Programmablauf können drei Optionen angegeben werden.

- Das Transportitem, das geändert werden soll
- Das zu schreibende Quellsystem (Idealerweise wird hier ein Quellsystem angegeben, das nicht wirklich existiert, es wird damit nur markiert, dass es sich nicht um das aktuelle System handelt.)
- Ein Flag zum Trockenlauf (wenn ein X gesetzt ist, dann wird keine Änderung vorgenommen sondern lediglich die Informationen zum gewählten Item angezeigt)

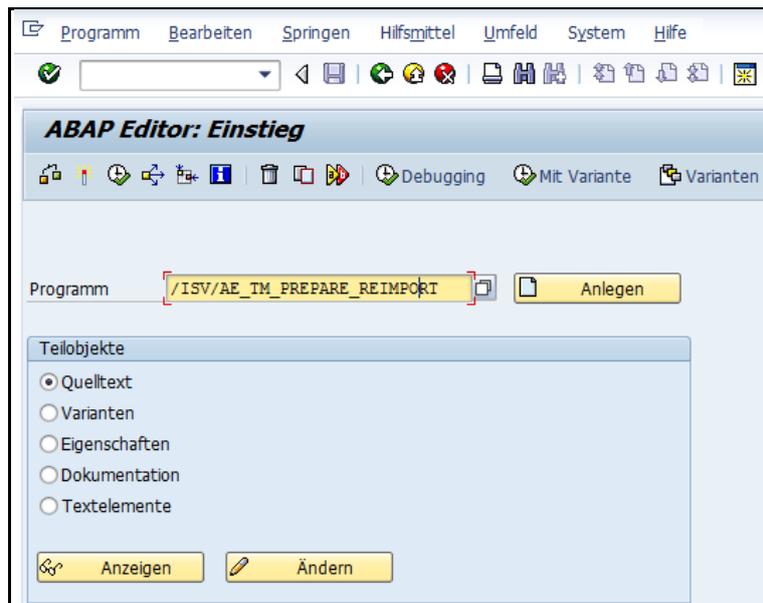


Abbildung40: Transportmanager - Programm um Reimport vorzubereiten



Abbildung41: Transportmanager - Optionen für Reimportvorbereitung

Nachdem die Quellsystem-ID mit dem genannten Programm geändert wurde, wird das Transportitem in der Liste der eingehenden Transporte angezeigt und kann normal importiert werden.

#### 9.4 Altes Transportvorgehen / vollständiger Transport

Für einen ersten Transport oder einen vollständigen Transport können alternativ auch alle Tabelleninhalte transportiert werden. Bitte beachten Sie dabei, dass die Views damit dann nicht als Transportiert markiert werden. Der Transport muss hierbei die folgenden Tabellen (vollumfänglich, also \*) beinhalten:

```

/ISV/AE_MDV_LANG
/ISV/AE_SETTING
/ISV/PAQ0DTX_AVC
/ISV/PAQ0DTX_MD1
/ISV/PAQ0DTX_MDV
/ISV/PAQ0DTX_UES
/ISV/PAQ0DTX_VD
/ISV/PAQ0DTX_VRA
/ISV/PAQ0DTX_VU

```

**Hinweis:** Die beiden Tabellen des Transportanschlusses (/IST/AE\_TRANSIT und /ISV/AE\_TRANSLOG) werden für einen Transport der AGIMENDO.edit-Installation nicht benötigt. Wenn allerdings das vollständige Produkt in der „aktuellen Situation“ gesichert werden soll, müssen diese Tabellen ebenfalls eingeschlossen werden.

**Hinweis:** Beim Transport aller Inhalte einer Tabelle (\*) werden die Inhalte dieser Tabelle im Zielsystem ersetzt, d.h. zunächst gelöscht und neu befüllt.

## 10.Anhang

Im ersten Unterkapitel befindet sich die HowTo-Sammlung zur Vorstellung von beispielhaften User-Exit-Implementierungen. Im Anschluss daran folgen die „technischen Kapitel“ zur Dokumentenhistorie, Abbildungsverzeichnis etc.

### 10.1 HowTo-Sammlung zur UserExit-Implementierung

#### 10.1.1 Anlage mit Vorlage

AGIMENDO.edit bringt eine allgemeine Implementierung mit um bei einer Neuanlage eines Satzes die Werte aus dem gerade ausgewählten Satz zu übernehmen. Dazu muss für den UserExit GET\_VALUE die Klasse /ISV/CL\_AE\_EXIT\_NEW\_TEMPLATE konfiguriert sein und alle Attribute, die aus dem ausgewählten Satz übernommen werden sollen, im Wizard in der Spalte „UE-Anlage“ markiert sein.

#### 10.1.2 Ausschluss von Merkmalen von der Gültigkeitsprüfung

Um ein Merkmal von der Gültigkeitsprüfung auszuschließen sind die folgenden Schritte zu durchlaufen:

- Anlage einer User-Exit-Klasse (nur wenn bisher UserExit nicht genutzt)
- Konfiguration der angelegten User-Exit-Klasse (nur wenn UserExit bisher nicht genutzt)
- Überschreiben/Erweitern der Methode TEST\_FOR\_EXCLUDE\_FROM\_CHECK
- Konkreter Ausschluss des gewünschten Merkmals

#### Anlage einer User-Exit-Klasse

Die UserExit Klasse wird nur einmal pro System benötigt. Wenn bisher keine Implementierung angelegt wurde, muss eine neue Klasse gemäß der für dieses System gültigen Namenskonventionen angelegt werden. Diese Klasse muss von der Klasse /ISV/PAQ0DCX\_UE\_IMP erben.

In Transaktion SE24 muss man den Klassennamen angeben und auf „Anlegen“ drücken. Im sich öffnenden Dialogfenster muss man auf das Icon hinter dem Klassennamen klicken um die gewünschte Vererbung hinzuzufügen. (Siehe Abbildung42)

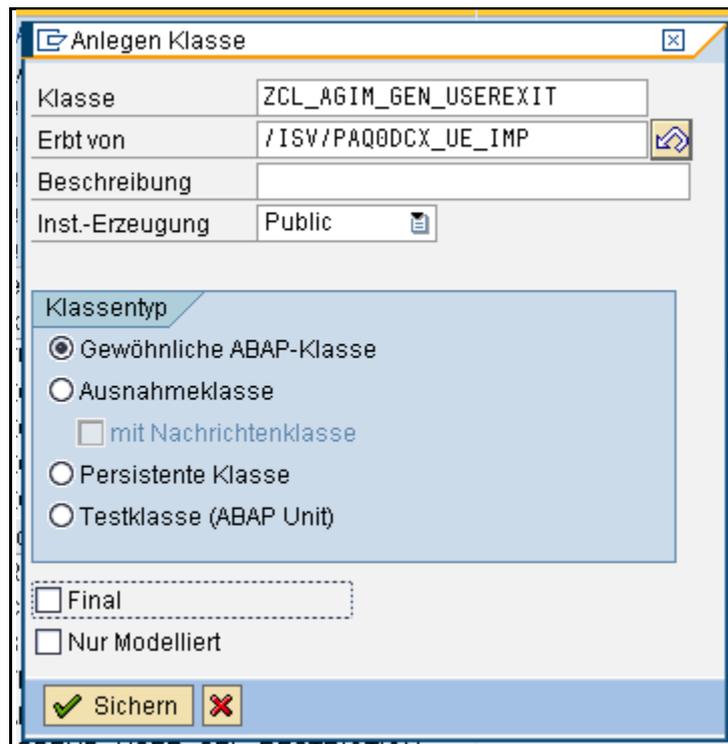


Abbildung42: Anlage UserExit Klasse

#### Konfiguration der angelegten User-Exit-Klasse

Die Überprüfung der gültigen Werte findet im Default-Coding des UserExits CUSTOM\_VALIDATION statt. Durch die Vererbung bleibt diese Funktionalität erhalten, in diesem HowTo wird lediglich eine Prüfmethode erweitert, die testet, ob ein Merkmal im Kontext eines InfoObjekts und Views überprüft werden soll oder nicht.

Dazu muss der UserExit CUSTOM\_VALIDATION der neu angelegten Klasse konfiguriert und aktiviert werden. Dies geschieht durch einen Eintrag in der Tabelle /ISV/PAQ0DTX\_UES. Jede Implementierung eines UserExits ist projektbezogen, d.h. sie bezieht sich auf einen konkreten View. Dafür wird neben dem Namen des UserExits und der Klasse, wo dieser UserExit implementiert wird, auch der Name des Projektes eingetragen, für das diese UserExit-Implementierung gilt. Um einen UserExit zu aktivieren, muss das Flag „ACTIVE“ in der Tabelle /ISV/PAQ0DTX\_UES gesetzt sein.

Das Hinzufügen dieses Eintrages erfolgt über die Transaktion SE16. Der Eintrag muss analog der Darstellung in Abbildung43 erfolgen.

**Data Browser: Tabelleninhalt /ISV/PAQ0DTX\_UES:**

MANDT	USEREXIT_NAME	PROJECT	CLASSNAME	ACTIVE
100	CUSTOM_VALIDATION	TEST_PROJECT	ZCL_AGIM_GEN_USEREXIT	X

Abbildung43: Konfiguration UserExit in Tabelle /ISV/PAQ0DTX\_UES

### Überschreiben/Erweitern der Methode TEST\_FOR\_EXCLUDE\_FROM\_CHECK

In der angelegten Klasse muss in der Methode TEST\_FOR\_EXCLUDE\_FROM\_CHECK der Ausschluss der Merkmale vorgenommen werden. Zu diesem Zweck muss, falls noch nicht geschehen, die Methode TEST\_FOR\_EXCLUDE\_FROM\_CHECK redefiniert werden. Wahlweise über die Transaktion SE80 oder SE24 kann die angelegte Klasse geöffnet werden und in der Liste der Methoden die Methode ausgewählt und über den Button „Redefinieren“ eine neue lokale Implementierung angelegt werden.

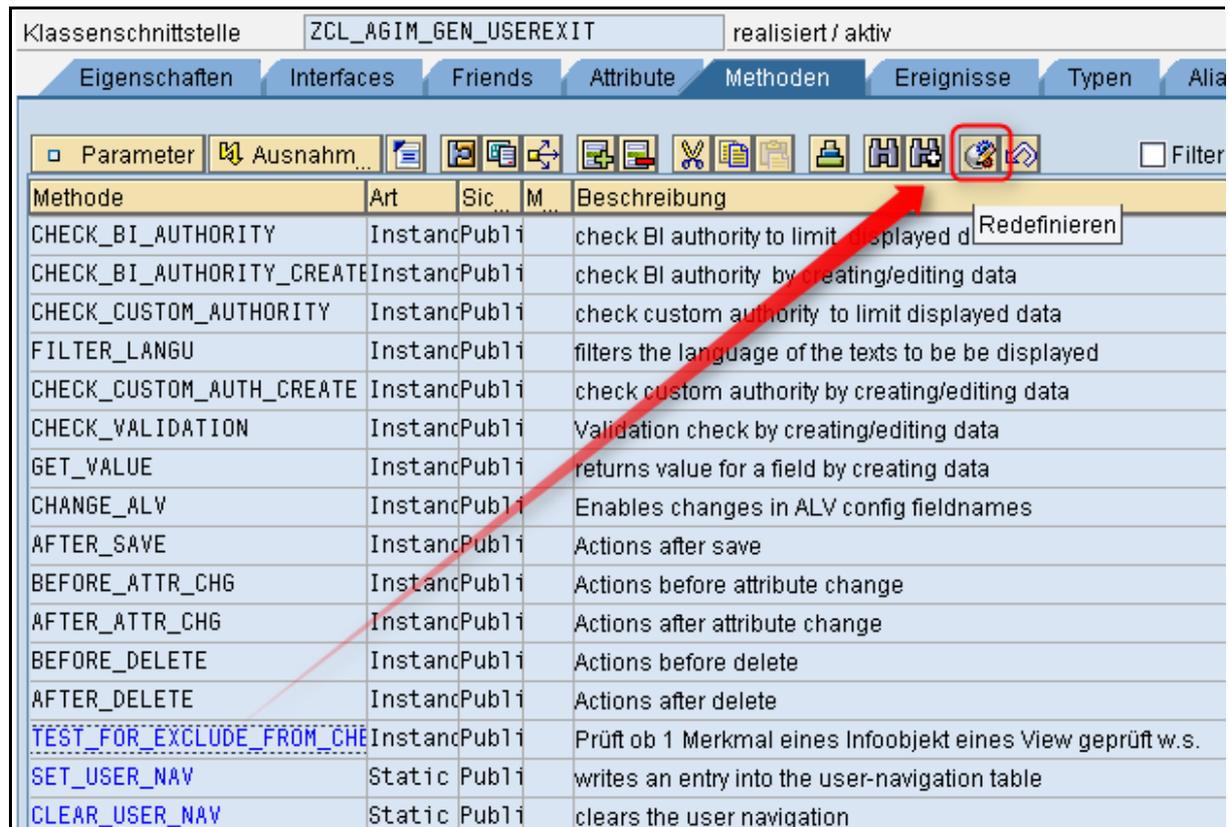


Abbildung44: Redefinition der Methode TEST\_FOR\_EXCLUDE\_FROM\_CHECK (SE80)

### Konkreter Ausschluss des gewünschten Merkmals

Nachdem die Erweiterung vorgenommen wurde, kann das gewünschte Merkmal im Coding der angelegten Redefinition der Methode TEST\_FOR\_EXCLUDE\_FROM\_CHECK ausgeschlossen werden. Dies geschieht dadurch, dass die Methode als Ergebniswert (Wert der Rückgabeveriable rv\_exclude) ein ‚X‘ zurück liefert. Als Eingabeparameter stehen die View-Nummer, der InfoObjekt-Name und der Spaltenname zur Verfügung.

Im einfachsten Fall könnte die Methode wie folgt implementiert werden:

```
method TEST_FOR_EXCLUDE_FROM_CHECK.
  rv_exclude = '-'.
  if iv_component = '/B135/S_CUSTOMER'.
    rv_exclude = 'X'.
  endif.
endmethod.
```

Nach dem Aktivieren steht der UserExit sofort zur Verfügung und wird durch AGIMENDO.edit aufgerufen.

Der Name, der für das Attribut hier verwendet werden muss (im Beispiel /B135/S\_CUSTOMER) entspricht dem Feldnamen, der im Wizard von AGIMENDO.edit angezeigt wird. Dies ist in der folgenden Abbildung45 dargestellt.

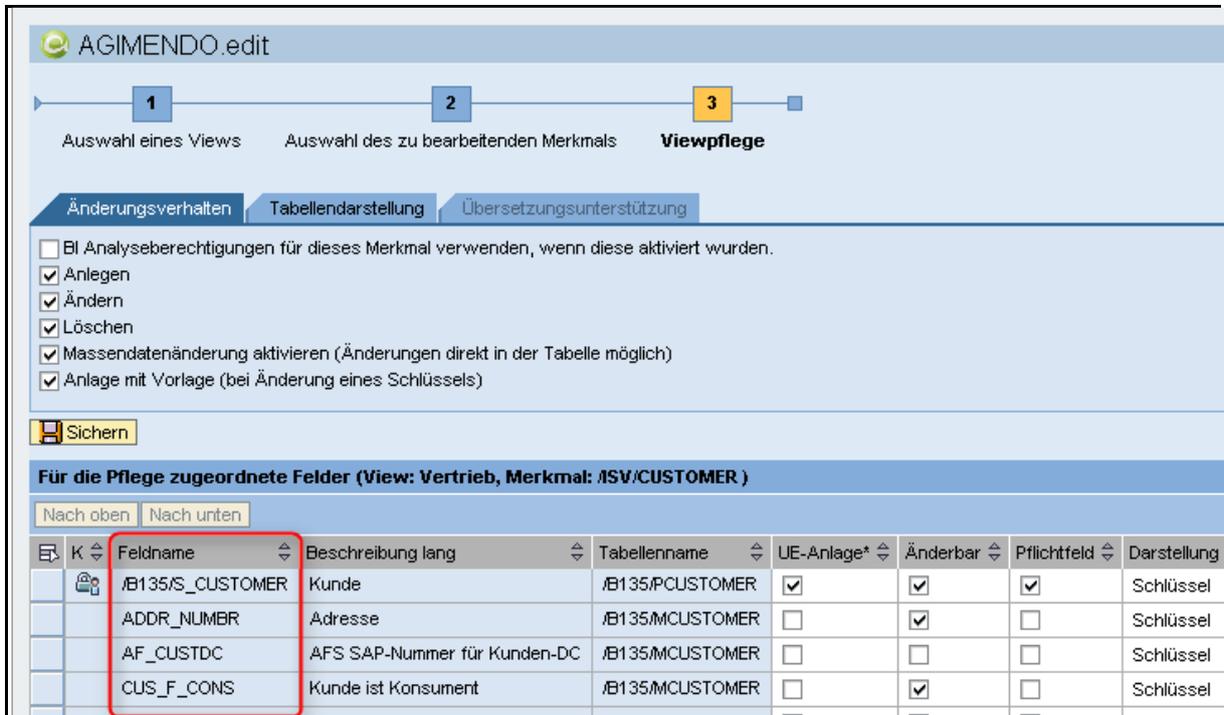


Abbildung45: Darstellung Feldnamen im AGIMENDO.edit Wizard

### 10.1.3 Ändern der Sortierspalten einer Tabelle

#### Überschreiben der Methode GET\_SORT\_FULL\_TABLE

Um die Sortierung einer Tabelle zu ändern muss die Methode GET\_SORT\_FULL\_TABLE redefiniert werden. Dabei kann für jedes InfoObject und jede View Nummer eine eigene Sortierung festgelegt werden.

#### Konkrete Festlegung der Sortierung

Im unten stehenden Beispiel wird für das InfoObject /ISV/PSS0WC04 festgelegt, dass die Tabelle nach dem Kurztext sortiert werden soll. Dazu muss auf Basis der Eingabeparameter iv\_iobj und iv\_view\_number das InfoObject und der View Nummer Einträge zur Rückgabetabelle rt\_sort\_table hinzugefügt werden.

Ein Eintrag in der Tabelle besteht aus dem Namen der Spalte, in diesem Fall dem Kurztext (TXTSH), der Sortierung (Aufsteigend: ' , bzw. Absteigend ,X') und der Sortiermethode (als Text ,X') festgelegt werden und der Rückgabetabelle hinzugefügt werden. Der Feldname, sowie der InfoObject Name und die View Nummer können dem AGIMENDO.edit Wizard entnommen werden.

```
method GET_SORT_FULL_TABLE.
  DATA lt_sort_table TYPE ABAP_SORTORDER_TAB.
  DATA ls_sort_table_entry TYPE ABAP_SORTORDER.

  IF iv_iobj EQ '/ISV/PSS0WC04' AND iv_view_number EQ 331.
```

```

ls_sort_table_entry-name = 'TXTSH'.
ls_sort_table_entry-descending = ' '.
ls_sort_table_entry-astext = 'X'.
APPEND ls_sort_table_entry TO lt_sort_table.
ENDIF.

rt_sort_table = lt_sort_table.

endmethod.

```

## 10.2 Referenz der UserExit

### 10.2.1 AFTER\_ATTR\_CHG (Methode: AFTER\_ATTR\_CHG)

Der UserExit AFTER\_ATTR\_CHG wird nach dem Starten des Attributänderungslaufs aufgerufen. Bei jedem Aufruf der Speichern-Funktionalität wird der Attributsänderungslauf gestartet, also auch beim Löschen und Neuanlegen eines Datensatzes.

Parameter	Typ	Beschreibung
IV_VIEW_NUMBER	INT2	View-Nummer

### 10.2.2 AFTER\_DELETE (Methode: AFTER\_DELETE)

Der UserExit AFTER\_DELETE wird nach dem erfolgreichen Löschen eines Datensatzes aufgerufen.

Parameter	Typ	Beschreibung
IT_COMPONENTS	/ISV/PAQ0DTX_WDR_FLDLABELS	Liste der Felder des Eintrags
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_VIEW_NUMBER	INT2	View-Nummer
IS_ATTR	TYPE REF TO DATA	Referenz zur ausgewählten Zeile

**Hinweis:** Nachdem der Datensatz gelöscht wurde, wird eine Änderung durch den UserExit an den Inhalten von IS\_ATTR nicht weitere beachtet.

### 10.2.3 AFTER\_SAVE (Methode: AFTER\_SAVE)

Der UserExit AFTER\_SAFE wird nach einem erfolgreichen Speichern der Attribute (und wenn notwendig Texte) aufgerufen. Der Parameter IV\_IS\_CREATED unterscheidet aufrufe aus einer Neuanlage heraus von „normalen“ Speicher-Aufrufen.

Parameter	Typ	Beschreibung
IT_COMPONENTS	/ISV/PAQ0DTX_WDR_FLDLABELS	Liste der Felder des Eintrags
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_VIEW_NUMBER	INT2	View-Nummer
IV_IS_CREATED	BOOLEAN	X=Eintrag wurde neu angelegt, Leer sonst.
IS_ATTR	TYPE REF TO DATA	Referenz zur ausgewählten Zeile

### 10.2.4 BEFORE\_ATTR\_CHG (Methode: BEFORE\_ATTR\_CHG)

Der UserExit BEFORE\_ATTR\_CHG wird vor einem Attributsänderungslauf aufgerufen.

Parameter	Typ	Beschreibung
IV_VIEW_NUMBER	INT2	View-Nummer

<b>EV_ERROR_OCCURED</b>	ABAP_BOOL	X=Es ist ein Fehler aufgetreten.
-------------------------	-----------	----------------------------------

**Hinweis:** in der aktuellen Implementierung wird der Parameter `EV_ERROR_OCCURED` ignoriert.

### 10.2.5 BEFORE\_DELETE (Methode: BEFORE\_DELETE)

Der UserExit `BEFORE_DELETE` wird vor einem Löschen eines Eintrages aufgerufen. Über den Parameter `EV_ERROR_OCCURED` kann der Datensatz vor dem löschen bewahrt werden. Eine Änderung an den Inhalten von `IS_ATTR` führt zu einer Änderung der Daten innerhalb der Anzeige, die aber erst beim nächsten Speichern-Aufruf auch gespeichert werden.

Parameter	Typ	Beschreibung
<b>IT_COMPONENTS</b>	/ISV/PAQ0DTX_WDR_FLDLABELS	Liste der Felder des Eintrags
<b>IV_IOBJ</b>	RSIOBJNM	InfoObjekt
<b>IV_VIEW_NUMBER</b>	INT2	View-Nummer
<b>IR_MESSAGE_MANAGER</b>	TYPE REF TO IF_WD_MESSAGE_MANAGER	Nachrichtenmanager zur Ausgabe von Nachrichten
<b>EV_ERROR_OCCURED</b>	ABAP_BOOL	X=Eintrag soll nicht gelöscht werden
<b>IS_ATTR</b>	TYPE REF TO DATA	Referenz zur ausgewählten Zeile

### 10.2.6 BI\_AUTHORISATION (Methode: CHECK\_BI\_AUTHORITY)

Der UserExit `BI_AUTHORISATION` wird aufgerufen, bevor die Daten für einen View geladen werden. Der UserExit ermittelt die Einschränkungen, die sich aus den BW-Analyseberechtigungen ergeben und erzeugt eine Bedingung die bei der Selektion der Daten angewandt wird. Dabei wird überprüft ob in der Definition des Views die Analyseberechtigungen aktiviert sind.

Parameter	Typ	Beschreibung
<b>IV_IOBJ</b>	RSIOBJNM	InfoObjekt
<b>IV_VIEW_NUMBER</b>	INT2	View-Nummer
<b>IV_USER</b>	XUBNAME	Aktueller Benutzer
<b>EV_BI_RESTRICTED</b>	CHAR1	„X“ = Es liegt eine Einschränkung vor und der Inhalt aus <code>EV_BI_RESTRICTION</code> soll beachtet werden.
<b>EV_BI_RESTRICTION</b>	STRING	Einschränkung, formuliert als Where- Bedingung eines Select-Statements auf der M-Tabelle des InfoObjekts.

**Hinweis:** Wenn dieser UserExit überschrieben wird, kann eine eigene Logik bei der Interpretation der Analyseberechtigungen angewandt werden. Die Standardimplementierung wird durch eine eigene Implementierung dieses UserExits überschrieben und muss explizit aufgerufen werden (`SUPER`-Aufruf).

**Hinweis:** Für eigene Bedingungen zur Datenfilterung sollte der Exit `CUSTOM_AUTORISATION` verwendet werden.

**Hinweis:** Der Wert `#` gilt standarderweise als ‚nicht zugewiesen‘.

### 10.2.7 BI\_AUTH\_CREATE (Methode: CHECK\_BI\_AUTHORITY\_CREATE)

Der UserExit BI\_AUTH\_CREATE wird aufgerufen um zu prüfen, ob ein konkreter Datensatz angelegt werden darf. Der UserExit ermittelt die Einschränkungen, die sich aus den BW-Analyseberechtigungen ergeben und testet den aktuellen Datensatz gegen diese Einschränkungen. Dabei wird überprüft ob in der Definition des Views die Analyseberechtigungen aktiviert sind. Der Exit wird auch bei einer Änderung der Attribute aufgerufen. (Dabei wird ja eine neue Kombination angelegt.)

Parameter	Typ	Beschreibung
IT_COMPONENTS	/ISV/PAQ0DTX_WDR_FLDLABELS	Liste der Felder des Eintrags
IR_MESSAGE_MANAGER	TYPE REF TO IF_WD_MESSAGE_MANAGER	Nachrichtenmanager zur Ausgabe von Nachrichten
IS_ATTR	TYPE REF TO DATA	Referenz zur ausgewählten Zeile
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_VIEW_NUMBER	INT2	View-Nummer
IV_USER	XUBNAME	Aktueller Benutzer
EV_ERROR_OCCURED	ABAP_BOOL	X=Eintrag soll nicht gelöscht werden

**Hinweis:** Wenn dieser UserExit überschrieben wird, kann eine eigene Logik bei der Interpretation der Analyseberechtigungen angewandt werden. Die Standardimplementierung wird durch eine eigene Implementierung dieses UserExits überschrieben und muss explizit aufgerufen werden (SUPER-Aufruf).

**Hinweis:** Für eigene Bedingungen zur Anlageüberprüfung sollte der Exit CUSTOM\_AUTH\_CREATE verwendet werden.

### 10.2.8 CHANGE\_ALV (Methode: CHANGE\_ALV)

Der UserExit CHANGE\_ALV wird bei jedem Round-Trip aufgerufen, also bei jeder Aktion oder Überprüfung der Daten. In diesem UserExit kann die Darstellung der Datentabelle angepasst werden. Vor allem können Spalten umbenannt, ausgeblendet, Reihenfolgen geändert etc. werden. Der UserExit wird für die gesamte Datentabelle einmal aufgerufen (nicht satzbasiert).

Parameter	Typ	Beschreibung
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_VIEW_NUMBER	INT2	View-Nummer
EV_TABLE_TITLE	STRING	Titel der Tabelle (Default wenn leer)
IT_CONFIG_TABLE	TYPE REF TO CL_SALV_WD_CONFIG_TABLE	Konfigurationsobjekt der Datentabelle

### 10.2.9 PFCG\_AUTHORISATION (Methode: CHECK\_PFCG\_AUTHORITY)

Der UserExit PFCG\_AUTHORISATION überprüft die Berechtigungen auf PFCG Ebene für das gerade zu bearbeitende InfoObjekt. Dabei wird je nach Aktion eine Überprüfung durchgeführt.

Parameter	Typ	Beschreibung
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_VIEW_NUMBER	INT2	View-Nummer
IV_USER	XUBNAME	Aktueller Benutzer
IV_ACTION	/ISV/AE_ACTION	VIEW   MAINTIAIN   DELETE   CREATE

<b>IR_MESSAGE_MANAGER</b>	TYPE REF TO IF_WD_MESSAGE_MANAGER	Nachrichtenmanager zur Ausgabe von Nachrichten
<b>EV_PFCG_RESTRICTED</b>	BOOLEAN	„X“ = Aktion ist nicht erlaubt

**Hinweis:** Wenn dieser UserExit überschrieben wird, kann eine eigene Logik zum Umgang mit den PFCG Berechtigungen angewandt werden. Die Standardimplementierung wird durch eine eigene Implementierung dieses UserExits überschrieben und muss explizit aufgerufen werden (SUPER-Aufruf).

### 10.2.10 CUSTOM\_AUTHORISATION (Methode: CHECK\_CUSTOM\_AUTHORITY)

Der UserExit CUSTOM\_AUTHORISATION bildet eine „eigene“ Berechtigungslogik ab. Dabei werden die Daten so gefiltert, dass der Benutzer nur Daten zu sehen bekommt, die er auch sehen darf. Der Exit kann natürlich auch dazu verwendet werden, die Menge der sichtbaren Daten anhand von Kriterien sinnvoll einzuschränken. Dabei wird eine Where-Bedingung für die Selektion auf der Stammdatentabelle erzeugt.

Parameter	Typ	Beschreibung
<b>IV_IOBJ</b>	RSIOBJNM	InfoObjekt
<b>IV_USER</b>	XUBNAME	Aktueller Benutzer
<b>IV_VIEW_NUMBER</b>	INT2	View-Nummer
<b>IV_TABNAME</b>	STRING	Name der Tabelle, gegen die die Selektion durchgeführt wird
<b>EV_CONDITION</b>	STRING	Einschränkung, formuliert als Where-Bedingung eines Select-Statements auf der M-Tabelle des InfoObjekts (Parameter IV_TABNAME).

### 10.2.11 CUSTOM\_AUTH\_CREATE (Methode: CHECK\_CUSTOM\_AUTH\_CREATE)

Der UserExit CUSTOM\_AUTH\_CREATE überprüft ob ein Datensatz angelegt werden darf. Dabei können neben eigenen Berechtigungslogiken auch inhaltliche Prüfungen der Daten erfolgen. Der UserExit wird satzweise aufgerufen, d.h. wenn mehrere Sätze angelegt/geändert wurden, wird der Exit für jeden Datensatz einmal aufgerufen.

Parameter	Typ	Beschreibung
<b>IT_COMPONENTS</b>	/ISV/PAQ0DTX_WDR_FLDLABELS	Liste der Felder des Eintrags
<b>IR_MESSAGE_MANAGER</b>	TYPE REF TO IF_WD_MESSAGE_MANAGER	Nachrichtenmanager zur Ausgabe von Nachrichten
<b>IS_ATTR</b>	TYPE REF TO DATA	Referenz zur ausgewählten Zeile
<b>IV_IOBJ</b>	RSIOBJNM	InfoObjekt
<b>IV_VIEW_NUMBER</b>	INT2	View-Nummer
<b>IV_USER</b>	XUBNAME	Aktueller Benutzer
<b>EV_CONDITION</b>	STRING	Einschränkung, formuliert als Where-Bedingung eines Select-Statements auf der M-Tabelle des InfoObjekts (Parameter IV_TABNAME).

### 10.2.12 CUSTOM\_POPUP\_FIELD (Methode: CUSTOM\_POPUP\_FIELD)

Der UserExit CUSTOM\_POPUP\_FIELD wird aufgerufen, wenn ein PopUp-Fenster erzeugt wird. Dies geschieht bei der Einzeländerung, bei der Neuanlage und beim „Ausgewählte Ändern“-Dialog. Der Exit wird für jedes Feld, das in dem PopUp dargestellt wird genau einmal durchlaufen und fragt ab, ob das Feld angezeigt werden soll und welcher Text dort angezeigt werden soll sowie ob das Feld eingabebereit ist. Für die CREATE Aktion ist in IR\_ATTR die gefüllte Zeile nach dem Ausführen des UserExit GET\_VALUE; für alle andere Aktionen ist in IR\_ATTR die ausgewählte Zeile des Views.

Parameter	Typ	Beschreibung
IV_VIEW_NUMBER	INT2	View-Nummer
IV_IOBJNM	RSIOBJNM	InfoObjekt
IV_COMPONENT	RSFIELDNM	Feldname
IV_POPUP	STRING	Art des PopUp: CHANGE (Einzeländerung)   CREATE (Neuanlage)   MASSCHANGE (Ausgewählte Ändern)
IR_ATTR	TYPE REF TO DATA	Referenz auf die ausgewählte Zeile
EV_HIDE_FIELD	BOOLEAN	„X“ = Feld soll ausgeblendet werden
CV_TEXT	STRING	Label-Text des Feldes
CV_READ_ONLY	BOOLEAN	„X“= Feld ist nur lesend in der Anzeige

### 10.2.13 CUSTOM\_VALIDATION (Methode: CHECK\_VALIDATION)

Der UserExit CUSTOM\_VALIDATION überprüft den Inhalt der Attributswerte des geänderten Eintrags. Das Default-Coding, das in dieser Methode aufgerufen wird, stellt sicher, dass für Merkmalswerte nur gültige InfoObjekt-Ausprägungen verwendet werden dürfen.

Parameter	Typ	Beschreibung
IR_MESSAGE_MANAGER	TYPE REF TO IF_WD_MESSAGE_MANAGER	Nachrichtenmanager zur Ausgabe von Nachrichten
IV_COMPONENTS	/ISV/PAQ0DTX_WDR_FLDLABELS	Tabelle der Felder
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_VIEW_NUMBER	INT2	View-Nummer
EV_ERROR_OCCURED	ABAP_BOOL	„X“ = Es ist ein Fehler aufgetreten
IS_ATTR	TYPE REF TO DATA	Referenz zur ausgewählten Zeile

**Hinweis:** Wenn dieser UserExit überschrieben wird, kann eine eigene Logik zum Umgang zur Werteüberprüfung angewandt werden. Die Standardimplementierung wird durch eine eigene Implementierung dieses UserExits überschrieben und muss explizit aufgerufen werden (SUPER-Aufruf).

**Hinweis:** Die Überprüfung der gültigen InfoObjektswerte ist in der Methode CHECK\_IOBJ\_VALIDATION der Klasse /ISV/PAQ0DCX\_UE\_IMP gekapselt. Wenn der UserExit von dieser Klasse erbt, kann die bestehende Logik einfach über den Aufruf dieser Methode weiter genutzt werden.

**Hinweis:** Wenn die Implementierung aus /ISV/PAQ0DCX\_UE\_IMP=> CHECK\_IOBJ\_VALIDATION genutzt wird, kann über eine zusätzliche Hook-Methode (Methode der Klasse /ISV/PAQ0DCX\_UE\_IMP, überschreibbar) TEST\_FOR\_MANDATORY\_FIELD\_CHECK eine Überprüfung einzelner Felder ausgeschlossen werden. Dazu muss diese Methode redefiniert werden und für jede von der Überprüfung auszuschließende Kombination aus (View /

*InfoObjekt / Attribut) ein „X“ zurückgegeben werden. Dies wird exemplarisch im Kapitel 10.1.2 Ausschluss von Merkmalen von der Gültigkeitsprüfung vorgestellt.*

### 10.2.14 EXECUTE\_BTN\_ACTION (Methode: EXECUTE\_BTN\_ACTION)

Der UserExit EXECUTE\_BTN\_ACTION wird ausgeführt, wenn ein benutzerspezifischer Button geklickt wurde (Definition benutzerspezifischer Buttons mit dem UserExit SHOW\_BUTTONS). Dabei wird der Exit satzweise für jeden markierten Eintrag aufgerufen. Es ist zwischen zwei Arten von Implementierungen zu unterscheiden: Zum einen kann eine ABAP-Funktionalität auf dem aktuell ausgewählten Datensatz durchgeführt werden und zum anderen kann ein Implementierungsspezifisches PopUp-Fenster (WebDynpro) angetriggert werden und angezeigt werden.

Parameter	Typ	Beschreibung
IV_VIEW_NUMBER	INT2	View-Nummer
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_BUTTON_ID	STRING	ID des geklickten Buttons
IV_LOCATION	/ISV/AE_BUTTON_LOCATION	BUTTON_BAR   SINGLE_CREATE_POPUP
IS_ATTR	TYPE REF TO DATA	Referenz zur ausgewählten Zeile
IR_MESSAGE_MANAGER	TYPE REF TO IF_WD_MESSAGE_MANAGER	Nachrichtenmanager zur Ausgabe von Nachrichten
EV_RESULT_STATUS	STRING	SHOW_WD_COMPONENT = WebDynpro-PopUp wird angezeigt, Leer sonst
ES_WD_TO_EXECUTE	/ISV/AE_WD_DESCRIPTOR	Struktur, die alle Informationen über das Anzuzeigende WebDynpro beinhaltet
EV_LINES_TO_INSERT	INT2	Anzahl der Zeilen, die neu zum View hinzugefügt werden sollen

Wenn der Ergebnisparameter EV\_RESULT\_STATUS gleich „SHOW\_WD\_COMPONENT“ ist, wird das WebDynpro, das in der Struktur ES\_WD\_TO\_EXECUTE beschrieben wird, angezeigt. Dabei müssen folgende Informationen enthalten sein:

- COMPONENT\_NAME: WebDynpro-Komponente z.B.: ZWD\_DYN\_INTERFACE\_TEST
- VIEW\_NAME: InterfaceView der Komponente. Z.B.: W\_MAIN
- PLUG\_NAME: Plugname z.B.: DEFAULT / wenn nicht vorhanden wird dieser ignoriert.
- EXTENDED\_PLUG\_TYPE: FALSE (zur Zeit nicht verwendet)

In der folgenden Abbildung sind der zu verwendende Komponentename und der Interface-View der Komponente markiert. Der Screenshot stammt aus dem Objektbaum der SE80 (ABAP Workbench).

Object Name	Description
<ul style="list-style-type: none"> <li>▼ ZWD_DYN_INTERFACE_TEST</li> <li>• COMPONENTCONTROLLER</li> <li>▼ Component Interface <ul style="list-style-type: none"> <li>▶ Implemented Interfaces</li> <li>• INTERFACECONTROLLER</li> <li>▼ Interface Views <ul style="list-style-type: none"> <li>• W_MAIN</li> </ul> </li> </ul> </li> <li>▶ Views</li> <li>▶ Windows</li> <li>▶ Web Dynpro Applications</li> </ul>	Test for dynamic compone

Abbildung46: Komponentename und Interface-View in SE80

### 10.2.15 EXEC\_BTN\_ACTION\_CLOS (Methode: EXEC\_BTN\_ACTION\_CLOSED)

Der UserExit EXEC\_BTN\_ACTION\_CLOS wird aufgerufen, wenn ein Eintrag durch einen Benutzerdefinierten Button und dazugehöriges UI bearbeitet wurde. Hierbei kann also eine Nachverarbeitung auf dem Eintrag vorgenommen werden, wenn die Oberfläche geschlossen wurde.

Parameter	Typ	Beschreibung
IV_VIEW_NUMBER	INT2	View-Nummer
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_BUTTON_ID	STRING	ID des geklickten Buttons
IV_LOCATION	/ISV/AE_BUTTON_LOCATION	BUTTON_BAR   SINGLE_CREATE_POPUP
IS_ATTR	TYPE REF TO DATA	Referenz zur ausgewählten Zeile
IR_MESSAGE_MANAGER	TYPE REF TO IF_WD_MESSAGE_MANAGER	Nachrichtenmanager zur Ausgabe von Nachrichten
IV_BUTTON_RESULT	STRING	OK or CANCEL

IV\_BUTTON\_RESULT enthält dabei genau einen der beiden Werte OK (wenn das Fenster mit „Schließen“ geschlossen wurde oder zum nächsten Eintrag gesprungen wurde bzw. CANCEL wenn das Fenster mit Abbrechen geschlossen wurde.

### 10.2.16 CUSTOM\_VALUE\_HELP (Methode: GET\_CUSTOM\_VALUE\_HELP)

Der UserExit CUSTOM\_VALUE\_HELP wird zu zwei Zeitpunkten aufgerufen. Zum einen bei der erstmaligen Erstellung des Views bei der Pflege (Erstaufruf je Session). Hier wird ermittelt für welche Felder eine spezifische Werthilfe verwendet werden soll. Wenn dann eine der Werthilfen ausgeführt wird, wird diese Methode erneut aufgerufen und die Einstellungen zur Darstellung der Werthilfe verwendet. Beide Aufrufe sollten je Feld dasselbe Ergebnis zurückliefern.

Über den Exit CUSTOM\_VALUE\_HELP kann zum einen eine Werthilfe für ein Attribut aktiviert oder deaktiviert werden und zum anderen kann für ein Attribut die Verwendung von AGIMENDO.edit als Werthilfe verwendet werden. Dazu wird ein View definiert, der hier als Werthilfe angezeigt werden soll sowie festgelegt, welche Attributswerte aus dem eingebetteten View in den Hauptview bei einer Auswahl übertragen werden sollen. Im Standardfall wird der einzelne Primärschlüssel (der InfoObjektswert) übernommen.

Der UserExit wird für jedes Attribut einzeln aufgerufen.

Parameter	Typ	Beschreibung
IV_VIEW_NUMBER	INT2	View-Nummer
IV_IOBJNM	RSIOBJNM	InfoObjekt
IV_COMPONENT	STRING	Name des Attributs
IS_ATTR	TYPE REF TO DATA	Referenz zur ausgewählten Zeile
EV_IOBJNM	RSIOBJNM	InfoObjekt, das als Werthilfe verwendet werden soll
EV_VIEW_NUMBER	INT2	Nummer des Views, der als AGIMENDO.edit Werthilfe verwendet werden soll
EV_AE_VALUE_HELP	BOOLEAN	„X“ = AGIMENDO.edit wird als Werthilfe verwendet
ET_FIELD_MAPPING	/ISV/AE_VH_FIELD_MAPPING_T	Feldmapping von Werthilfe auf Hauptview (ATTR_SRC beschreibt den Namen des Attributs in der Werthilfe, ATTR_TAR den Namen des Attributs im Hauptview)
EV_OTR_TEXT	STRING	OTR-Alias für einen Hinweistext in der Werthilfe
EV_VH_DEACTIVATED	Boolean	„X“ = Werthilfe ist für dieses Attribut vollständig deaktiviert

**Hinweis:** Wenn kein Feldmapping angegeben ist (Tabelle ET\_FIELD\_MAPPING ist initial) wird der Name des Attributs im Hauptview sowohl als Quelle als auch als Ziel verwendet.

### 10.2.17 FILTER\_LANGU (Methode: FILTER\_LANGU)

Der UserExit FILTER\_LANGU wird analog dem UserExit CUSTOM\_AUTHORISATION dazu verwendet, die Menge der angezeigten Daten zu reduzieren. Allerdings bezieht sich dieser Exit auf die Sprachen der Texte. Der UserExit wird nicht aufgerufen, wenn die Übersetzungsunterstützung aktiviert ist.

Parameter	Typ	Beschreibung
IV_IOBJ	RSIOBJNM	InfoObjekt
IV_USER	XUBNAME	Aktueller Benutzer
IV_VIEW_NUMBER	INT2	View-Nummer
IV_TABNAME	STRING	Name der Tabelle, gegen die die Selektion durchgeführt wird (T-Tabelle)
EV_CONDITION	STRING	Einschränkung, formuliert als Where-Bedingung eines Select-Statements auf der T-Tabelle des InfoObjekts (Parameter IV_TABNAME).

### 10.2.18 GET\_VALUE (Methode: GET\_VALUE)

Der UserExit GET\_VALUE wird für jedes Attribut aufgerufen, das im Wizard für „UE-Anlage“ aktiviert ist. Der Rückgabewert des UserExit wird dann im PopUp-Fenster als neuer Wert hinterlegt. Dadurch ist eine Vorbelegung mit Werten möglich, bzw. können neue IDs generiert werden und müssen nicht vom Benutzer „erdacht“ werden.

Parameter	Typ	Beschreibung
I_INFOOBJECT	RSIOBJNM	InfoObjekt
I_ATTR	FIELDNAME	Name des Feldes, für das der Wert ermittelt werden soll
I_VIEW_NUMBER	INT2	View-Nummer

<b>IS_ATTR</b>	TYPE REF TO DATA	Referenz zur ausgewählten Zeile, initial, wenn keine Zeile ausgewählt
<b>I_MODE</b>	STRING	Methode oder Ort aus der GET_VALUE aufgerufen wird. Kann den Wert ‚CREATE‘ annehmen, wenn sie beim Erzeugen eines neuen Eintrags aufgerufen wird. Und ‚EMPTY_LINE‘, wenn nur eine neue leere Zeile angefordert wird (GET_NEW_LINES_COUNT).
<b>E_DATA</b>	ANY	Wert, der als Vorbelegungswert zurückgegeben werden soll

**Hinweis:** Es gibt eine UserExit-Implementierung, die Anlage mit Vorlage implementiert, die mit AGIMENDO.edit mit ausgeliefert wird. Die Klasse, die diesen Exit enthält, heißt /ISV/CL\_AE\_EXIT\_NEW\_TEMPLATE.

### 10.2.19 GET\_KEYS\_F\_SELECTION (Methode: GET\_KEYS\_F\_SELECTION)

Der UserExit GET\_KEYS\_F\_SELECTION liefert eine Liste von Feldern, die zusammen eine Zeile einer Tabelle semantisch eindeutig identifizieren. Über die Liste der zurückgegebenen Felder kann so nach einer Tabellenänderung für eindeutig identifizierte Zeilen die Auswahl auf diesen neu gesetzt werden. Die Defaultimplementierung nutzt die Datenbankschlüssel als semantische Schlüssel. Wenn diese bspw. beim Speichern neu erzeugt werden, muss der Exit überschrieben werden, damit die Tabellenauswahl über ein Neuladen erhalten bleiben kann.

Parameter	Typ	Beschreibung
<b>IV_VIEW_NUMBER</b>	/ISV/VIEW_NUMBER	View-Nummer
<b>IV_IOBJ</b>	RSIOBJNM	Name des InfoObjekts
<b>ET_KEY_COMPONENTS</b>	/ISV/PAQ0DTX_WDR_FLDLABELS	Liste der semantischen Schlüsselfelder

### 10.2.20 GET\_NEW\_LINES\_COUNT (Methode: GET\_NEW\_LINES\_COUNT)

Der UserExit GET\_NEW\_LINES\_COUNT liefert die Anzahl von leeren, nicht gespeicherten neuen Zeilen zurück, die an das Ende der Tabelle bei jedem Öffnen angehängt werden. Wird eine dieser neuen Zeilen geändert, wird diese beim Speichern als neue Zeile gespeichert. Mit dem UserExit GET\_VALUE können Werte in den Zeilen vorbelegt werden. (Insbesondere muss das bedacht werden, wenn IDs über Nummernkreise erstellt werden, da diese dann ggf. verloren gehen).

Parameter	Typ	Beschreibung
<b>IV_VIEW_NUMBER</b>	INT2	View-Nummer
<b>IV_IOBJNM</b>	RSIOBJNM	Name des InfoObjekts
<b>EV_COUNT</b>	INT2	Anzahl der einzufügenden Zeilen
<b>EV_MODE</b>	CHAR5	START = Zeilen werden am Anfang eingefügt, sonst: Zeilen werden am Ende eingefügt

### 10.2.21 GET\_SORT\_FULL\_TABLE (Methode: GET\_SORT\_FULL\_TABLE)

Der UserExit GET\_SORT\_FULL\_TABLE liefert die Vorsortierung der Tabelle zurück. Dabei kann über Einträge des Parameters RT\_SORT\_TABLE vom Typ ABAP\_SORTORDER\_TAB eine Liste von

Spalten angegeben werden, nach denen sortiert werden soll. Zusätzlich kann ein Flag EV\_DELETE\_ADJACENT gesteuert werden, ob Duplikate (gemessen an den ersten fünf Sortierfeldern) gelöscht werden sollen oder nicht. Der Exit wird einmal zu Beginn und bei jedem „Neuladen“ des View ausgeführt.

Parameter	Typ	Beschreibung
IV_IOBJ	RSIOBJNM	Name des InfoObjekts
IV_VIEW_NUMBER	INT2	View-Nummer
EV_DELETE_ADJACENT	BOOLEAN	„X“=Duplikate (gemessen an den bis zu ersten fünf Sortierfeldern) werden gelöscht
RT_SORT_TABLE	ABAP_SORTORDER_TAB	Sortierfelder, für jedes Feld die Information ob Aufsteigend oder absteigend und ob als Zahl oder Text

**Hinweis:** Wenn Duplikate gelöscht werden, werden diese natürlich nur aus der Anzeige nicht aus dem InfoObjekt gelöscht.

### 10.2.22 PREPAPRE\_PREVIEW\_TABLE (Methode: PREPARE\_PREVIEW\_TABLE)

Der UserExit PREPARE\_PREVIEW\_TABLE wird vor der Vorschau einer CSV oder XLSX Excel Datei aufgerufen. Dadurch wird dem User ermöglicht die Excel-Tabelle noch einmal vor dem Einordnen der Spalten und vor dem Upload in die View zu bearbeiten.

Parameter	Typ	Beschreibung
IV_IOBJ	RSIOBJNM	Name des InfoObjekts
IV_VIEW_NUMBER	INT2	View-Nummer
EV_ERROR_FLAG	BOOLEAN	„X“ = Es ist ein Fehler aufgetreten
ET_ERROR_STRING_TABLE	STRING_TABLE	Eine Liste mit Fehlermeldungen
CT_UPLOAD_TABLE	DATA	Die Tabelle die in die View hochgeladen wird

### 10.2.23 PREPAPRE\_UPLOAD\_TABLE (Methode: PREPARE\_UPLOAD\_TABLE)

Der UserExit PREPARE\_UPLOAD\_TABLE wird vor dem Upload einer CSV oder XLSX Excel Datei aufgerufen. Dadurch wird dem User ermöglicht die Upload-Tabelle noch einmal nach dem Einordnen der Spalten und vor dem Upload in die View zu bearbeiten.

Parameter	Typ	Beschreibung
IV_IOBJ	RSIOBJNM	Name des InfoObjekts
IV_VIEW_NUMBER	INT2	View-Nummer
EV_ERROR_FLAG	BOOLEAN	„X“ = Es ist ein Fehler aufgetreten
ET_ERROR_STRING_TABLE	STRING_TABLE	Eine Liste mit Fehlermeldungen
CT_UPLOAD_TABLE	DATA	Die Tabelle die in die View hochgeladen wird

### 10.2.24 READ\_ONLY\_MODE (Methode: READ\_ONLY\_MODE)

Dieser UserExit wird beim Öffnen eines Views durchlaufen und bietet die Möglichkeit an, einen View als „Read Only“ anzuzeigen. Dabei kann der View nur eingesehen werden und keine Änderungen an den Daten vorgenommen werden. Dieser Exit bietet sich für eine Sperrung analog einer Datenscheibe an.

Parameter	Typ	Beschreibung
IV_IOBJ	RSIOBJNM	Name des InfoObjekts
IV_VIEW_NUMBER	INT2	View-Nummer
IT_DATA_TABLE	DATA	Tabelle mit allen Inhalten
EV_READ_ONLY	Boolean	„X“ Entspricht nur lesender Zugriff
EV_READ_ONLY_MESSAGE	String	Nachricht, die als Begründung ausgegeben wird

### 10.2.25 SKIP\_ATTRIB\_CHANGE (Methode: SKIP\_ATTRIB\_CHANGE)

Der UserExit wird nach jeder Änderung oder Speicherung eines Datensatzes durchgelaufen und bietet die Möglichkeit, die Option um das Attributsänderungslauf (Konfiguration je View/InfoObjekt im Wizard) zu überspringen zu überschreiben.

Parameter	Typ	Beschreibung
IV_ACTION	STRING	SAVE   CHANGE   CREATE
IV_IOBJ	RSIOBJNM	Name des InfoObjekts
IV_VIEW_NUMBER	INT2	View-Nummer
CV_DEACTIVATED	BOOLEAN	„X“ = Die Attributsänderungslauf soll übersprungen werden

### 10.2.26 SHOW\_BUTTONS (Methode: GET\_UE\_BUTTONS\_FOR\_LOCATION)

Der UserExit SHOW\_BUTTONS liefert eine Liste von Buttons zurück, die an der definierten Stelle angezeigt werden sollen. Die Implementierung der Buttons (also die Logik die ausgeführt wird, wenn auf den Button geklickt wird) wird in dem UserExit EXECUTE\_BTN\_ACTION definiert.

Parameter	Typ	Beschreibung
IV_LOCATION	/ISV/AE_BUTTON_LOCATION	SINGLE_CREATE_POPUP   BUTTON_BAR
IV_IOBJ	RSIOBJNM	Name des InfoObjekts
IV_VIEW_NUMBER	INT2	View-Nummer
RT_USER_EXIT_BUTTONS	/ISV/AE_EXIT_BUTTON_TABLE_T	Tabelle mit ExitButtons

Die Tabelle RT\_USER\_EXIT\_BUTTONS enthält die folgenden Komponenten:

- BUTTON\_LOCATION: Ort des Buttons (SINGLE\_CREATE\_POPUP | BUTTON\_BAR)
- BUTTON\_ID: Eindeutige Button-ID, Identifizierung der Aktion über diese Button-ID
- BUTTON\_OTR\_TEXT: OTR-Text, der auf dem Button angezeigt werden soll
- BUTTON\_ICON: Icon, das auf dem Button angezeigt werden soll

- BUTTON\_POSITION:** Position, an der der Button platziert werden soll. Im PopUp bei der Anlage (**SINGLE\_CREATE\_POPUP**) gibt es die folgenden Positionen: BEFORE\_SAVE, BEFORE\_CANCEL, AFTER\_CANCEL. Jeder andere Wert ist gleichgesetzt mit AFTER\_CANCEL. Im Falle der Anzeige des Buttons in der Zeile unterhalb der Datentabelle (Position BUTTON\_BAR) gibt es die Positionen BEFORE\_CREATE, BEFORE\_MASS\_CHANGE, BEFORE\_DELETE, BEFORE\_TEMP\_SAVE, BEFORE\_SAVE, BEFORE\_CHANGE und AFTER\_CHANGE. Jeder andere Wert (auch leer) ist gleichgesetzt mit AFTER\_CHANGE. In der Folgenden Abbildung sind die Buttons in ihrer Reihenfolge dargestellt. Wenn bspw. über den Wizard das Löschen eines Datensatzes deaktiviert ist haben die beiden Positionen BEFORE\_DELETE und BEFORE\_TEMP\_SAVE für einen Button dasselbe Ergebnis. Allerdings werden zuerst alle Buttons mit der Position BEFORE\_DELETE und dann alle Buttons mit der Position BEFORE\_TEMP\_SAVE angezeigt.

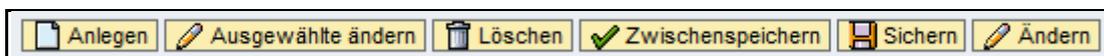


Abbildung 47: Buttons in der Massenpflege

Beispiel für die Implementierung dieses UserExits für die Anzeige eines eigenen Buttons vor allen anderen:

```
method GET_UE_BUTTONS_FOR_LOCATION.

  DATA ls_button LIKE LINE OF rt_user_exit_buttons.

  IF iv_location = 'BUTTON_BAR'.
    ls_button-button_location = 'BUTTON_BAR'.
    ls_button-button_id = 'MY_BUTTON_ID'.
    ls_button-button_otr_text = 'ZMYPACKAGE/BUTTON_LABEL'.
    ls_button-button_icon = 'ICON_DELETE'.
    "location: SINGLE_CREATE_POPUP
    "Order: BEFORE_SAVE <SAVE> BEFORE_CANCEL <CANCEL> anything else
    "location: BUTTON_BAR
    "Order BEFORE_CREATE <CREATE> BEFORE_MASS_CHANGE <MASS_CHANGE>
    " BEFORE_DELETE <DELETE> BEFORE_TEMP_SAVE <TEMP_SAVE>
    " BEFORE_SAVE <SAVE> BEFORE_CHANGE <CHANGE> anything else
    ls_button-button_position = 'BEFORE_CREATE'.
    APPEND ls_button TO rt_user_exit_buttons.
  ENDIF.

endmethod.
```

### 10.2.27 TEXT\_CHANGED (Methode: TEXT\_CHANGED)

Der UserExit TEXT\_CHANGED wird aufgerufen, wenn ein Text sich geändert hat. Dabei wird der Exit für jede Zielsprache der Übersetzungsunterstützung genau einmal aufgerufen und der jeweils geänderte Originaltext mitgegeben. Damit der Exit aufgerufen wird, muss im Wizard das Flag „Textänderungen an UserExit weiterleiten“ aktiviert sein.

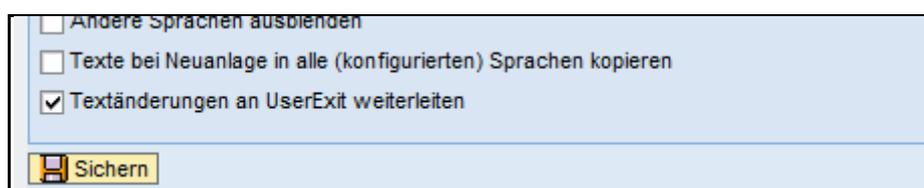


Abbildung 48: Konfiguration zum Typ  
UserExit

Beschreibung

TextänderungserkennungParameter		
<b>IV_VIEW_NUMBER</b>	INT2	View-Nummer
<b>IV_IOBJNM</b>	RSIOBJNM	Name des InfoObjekts
<b>IV_TARGET_LANGU</b>	SY-LANGU	Zielsprache in die übertragen werden soll
<b>EV_IGNORE_LANGUAGE</b>	BOOLEAN	„X“=Textübertragung wird für diese Sprache ignoriert (Default-Coding)
<b>CS_TEXT</b>	ANY	Struktur der Text-Tabelle, die eine Zeile des Textes, der auf die Zielsprache kopiert werden soll, darstellt.

## 10.3 API Methoden Beschreibung

### 10.3.1 GET\_CHANGES

Diese Methode erlaubt zu wissen, ob Änderungen des Kontexts zwischen zwei Zeitpunkte geschehen sind. Die Natur der Änderungen ist in der Struktur der Tabelle auch gespeichert.

Parameter	Typ	Beschreibung
<b>IV_AND_RESET_CHANGES</b>	BOOLEAN	“X” = Änderungslog hinterher löschen
<b>RT_CHANGES</b>	WDR_CONTEXT_CHANGE_LIST	Tabelle von alle Änderungen des Kontexts seit dem Anfang oder dem letzten Reset

### 10.3.2 GET\_READ\_ONLY

Mit dieser Methode kann wissen, ob das View gesperrt ist oder nicht. EV\_READ\_ONLY\_MSG ermöglicht, den Benutzer über den Grund der Sperrung zu informieren.

Parameter	Typ	Beschreibung
<b>EV_READ_ONLY</b>	WDY_BOOLEAN	“X” = View ist nur lesbar
<b>EV_READ_ONLY_MSG</b>	STRING	Meldung dem Grund der Sperrung

### 10.3.3 GET\_SELECTED\_ENTRIES

Diese Methode gibt zurück den Inhalt von alle ausgewählte Zeile im View.

Parameter	Typ	Beschreibung
<b>CT_SELECTED_ENTRIES</b>	HASHED TABLE	Liste von alle ausgewählte Zeilen im View

### 10.3.4 GET\_SELECTED\_ENTRY

Diese Methode gibt zurück den Inhalt der ausgewählten Zeile im View durch Lead Selection.

Parameter	Typ	Beschreibung
<b>ER_ENTRY</b>	DATA	Ausgewählte Zeile im View

### 10.3.5 GET\_URL\_PARAMS

Diese Methode gibt die Kombination InfoObjekt/View Nummer zurück, die den aktuellen View definiert.

Parameter	Typ	Beschreibung
EV_URL_PARAM_INFOOBJECT	STRING	Name des InfoObjekts
EV_URL_PARAM_VIEW_NUMBER	STRING	View-Nummer

### 10.3.6 RAISE\_EXTERNAL\_BUTTON\_EVENT

Diese Methode ermöglicht ein Klick auf einem Knopf des Views zu simulieren. Es gilt für alle Knöpfe Standard oder, der UserExit SHOW\_BUTTONS oder generierte Knöpfe in der View Tabelle.

Parameter	Typ	Beschreibung
IV_BUTTON_ID	STRING	ID des Knopfs zu aktivieren

### 10.3.7 READ\_TABLE\_ROW

Diese Methode gibt zurück den Inhalt der ausgewählten Zeile im View durch den gegebenen Index.

Parameter	Typ	Beschreibung
IV_INDEX	I	Index der Zeile zu lesen
RS_ATTR	DATA	Ausgewählte Zeile im View

### 10.3.8 SET\_ENTRY

Diese Methode ändert alle Felder von der durch Lead Selection ausgewählten Zeile mit den gegebenen Daten in IR\_ENTRY.

Parameter	Typ	Beschreibung
IR_ENTRY	DATA	Daten im View zu speichern

### 10.3.9 SET\_LEAD\_SELECTION

Diese Methode stellt das Lead Selection auf der ersten Zeile des View, die sich mit allen angegebenen Feldname-Wert Kombination deckt.

Parameter	Typ	Beschreibung
IT_SEMANTIC_KEYS	/ISV/AE_NAME_VALUE_T	Kombination Feldname-Wert zu finden für das Lead Selection
IV_CLEAR_LEAD_SELECTION	BOOLEAN	Selektion und Lead Selektion ist geklärt falls keine passende Zeile gefunden wurde

## 10.4 Änderungshistorie

Version	Autor	Änderung	Datum
1.0	Rouzbeh Nabatian	Erstellung und Zusammenführung	10.05.2011
1.1	Silke Jung	Änderungen nach Review	30.05.2011
1.2	Felix Weyde	Erweiterung um HowTo 9.1.1	13.07.2011
1.3	Sebastian Esch	Erweiterung um HowTo 9.1.2	11.06.2012

<b>1.4</b>	Sebastian Esch	Überarbeitung für Release 2.11, Parametrisierter Aufruf der Stammdatenpflege	21.09.2012
<b>1.4.1</b>	Sebastian Esch	Fehlerhafte Links zu den AGIMENDO.edit Komponenten korrigiert	16.10.2012
<b>1.4.2</b>	Artjom Smirnov	P. 7 „Berechtigungen“ vervollständigt	19.12.2012
<b>1.4.3</b>	Artjom Smirnov	UserExit Dokumentation	20.01.2013
<b>1.4.4</b>	Artjom Smirnov	Erweiterungen in Wizard	08.02.2013
<b>1.5.0</b>	Felix Weyde	Erweiterung der UserExit Dokumentation	25.03.2013
<b>1.6.0</b>	Felix Weyde	Transportmanagement	27.01.2014
<b>1.6.1</b>	Felix Weyde	Anpassung Buttonsichtbarkeit / UserExit-Doku	02.07.2014
<b>1.6.2</b>	Kemal Akman	UserExit-Verwaltung / UserExit Doku	06.08.2014
<b>1.6.3</b>	Felix Weyde	Anpassung zu Release 2.13.0	01.07.2015
<b>1.7.0</b>	Mathieu André	Anpassung zu Release 2.14.1 + API Dokumentation	05.04.2016

## 10.5 Abbildungsverzeichnis

Abbildung 1: Funktionsweise AGIMENDO.edit .....	6
Abbildung 2: Selektion des Views.....	7
Abbildung 3: Zuordnung von Benutzer bzw. Rollen zu einem View .....	8
Abbildung 4: Auswahl eines Views.....	9
Abbildung 5: Auswahl des zu bearbeitenden Merkmals.....	9
Abbildung 6: Ausgewähltes Merkmal bearbeiten.....	10
Abbildung 7: Attribute zuordnen .....	11
Abbildung 8: Viewpflege .....	12
Abbildung 9: Konfiguration des Pflegeviews (Änderungsverhalten) .....	12
Abbildung 10: Konfiguration des Pflegeviews (Tabellendarstellung) .....	14
Abbildung 11: Konfiguration des Pflegeviews (Übersetzungsunterstützung).....	15
Abbildung 12: Konfiguration des Pflegeviews (Spaltenerweiterungen) .....	17
Abbildung 13: Konfiguration des Pflegeviews (Technische Einstellungen).....	18
Abbildung 14: Konfiguration des Pflegeviews (Attribute definieren) .....	18
Abbildung 15: Startbildschirm Stammdatenpflege .....	20
Abbildung 16: Wirkung der im Wizard vorgenommenen Einstellungen auf die Pflegeoberfläche .....	21
Abbildung 17: Parametrisierter Aufruf der Stammdatenpflege .....	22
Abbildung 18: Definition und Aktivierung der UserExits.....	23
Abbildung 19: "UserExits Pflegen" Button .....	24
Abbildung 20: Übersicht des Fenster UserExits Verwaltung.....	25
Abbildung 23: User Exit bei der Auswahl des zu bearbeitenden Merkmals .....	28
Abbildung24: User Exit beim Anlegen von Datensätzen .....	29
Abbildung 25: User Exit beim Speichern von Datensätzen .....	29
Abbildung 26: Kein User Exit bei "Änderungen übernehmen" .....	30
Abbildung27: User Exit beim Zwischenspeichern von Datensätzen .....	30
Abbildung28: User Exit beim Löschen von Datensätzen .....	30
Abbildung29: Aktivitäten des Berechtigungsobjekts /ISV/AEDIT.....	31
Abbildung30:Optionen beim Transport eines Views .....	33
Abbildung31: Wizard - Anzeige von Transport-Status .....	34
Abbildung32: Wizard - Technische Einstellungen zum Transportwesen .....	35
Abbildung33:Wizardwarnung bei Bearbeitung im Nicht-Quellsystem .....	35

Abbildung34: Ausgehende Transporte im AGIMENDO.edit Transportmanager .....	36
Abbildung35:Transportmanager - Einstellungen zum Transportieren eines InfoObjekt/View .....	37
Abbildung36: Transportmanager - Einstellungen zum Transport eines Views .....	38
Abbildung37: Transportmanager - Einstellung zum Transport eines Projekts .....	39
Abbildung38: Transportmanager - Manueller Transport: Zuweisung eines Transportauftrags .....	39
Abbildung39: Transportmanager - Eingehende Transporte .....	40
Abbildung40: Transportmanager - Programm um Reimport vorzubereiten .....	41
Abbildung41: Transportmanager - Optionen für Reimportvorbereitung .....	41
Abbildung42: Anlage UserExit Klasse .....	44
Abbildung43: Konfiguration UserExit in Tabelle /ISV/PAQ0DTX_UES .....	44
Abbildung44: Redefinition der Methode TEST_FOR_EXCLUDE_FROM_CHECK (SE80) .....	45
Abbildung45: Darstellung Feldnamen im AGIMENDO.edit Wizard .....	46
Abbildung46: Komponentename und Interface-View in SE80.....	53
Abbildung 47:Buttons in der Massenflege.....	58
Abbildung48:Konfiguration zum UserExit Textänderungserkennung.....	58