

**Erfassung, Verwaltung und Dokumentation der
Mengenströme an Anlagen
zur Sortierung und Verwertung von Wertstoffen**

HANDBUCH WAAGESCHNITTSTELLE

Version 2.11.0154
Stand: Dezember 2016

1 INHALTSVERZEICHNIS

1 INHALTSVERZEICHNIS	3
1 Einleitung	6
1.1.1 Meldung	6
1.1.2 Transfer	6
1.1.3 Waageschnittstelle	6
1.1. Gegenstand dieses Dokumentes	7
1.2. Glossar	7
1.3. Liste der verwendeten Symbole	7
1.4. Übersicht	8
2 Import	10
2.1. Zusammenhänge	10
2.2. Grober Ablauf	11
2.2.1. Anlegen einer Routine	11
1.1.4 Feldbeschreibungen	12
2.2.2. Konfiguration der XML - Dateien	14
2.2.2.1. Beschreibung der -import.xml (WsBA31Beispiel-import.xml)	15
2.2.2.2. Beschreibung der -file-definition.xml (BA31-file-definition.xml)	16
2.2.2.3. Beschreibung der -interface.xml (BA31-interface.xml)	17
2.2.2.4. Verknüpfen von -interface.xml und -file-definition.xml	18
2.2.2.5. Multi - Mapping (Schlüsseldefinitionen)	19
2.2.2.6. Einbinden von Multi - Mappings in interface.xml	20
2.2.3. Import der Daten	21
2.3. Fahrplan Import	22
1.1.5 Oberfläche ^{wm} e.fact	22
1.1.6 XML-Editor	22
1.1.7 Oberfläche ^{wm} e.fact	22
3 Export	23
3.1. Allgemein	23
3.2. Anlegen einer Export-Task	23
1.1.8 Besonderheiten:	24
4 Nachbearbeitungsdialoge für den Import	25
4.1. Dialog: <i>Waage Schnittstelle Pflege</i>	25
4.2. Dialog: <i>Waage Schnittstelle Löschen</i>	26
1.1.9 Bereich \ Anzahl	26
1.1.10 Ohne Wiegeschein	26
1.1.11 Vollständig	26
4.3. Dialog: <i>Waage Schnittstelle Übernahme</i>	28
5 Beispiel für die Umsetzung der alten Meldeschnittstelle BA31/32	29
1.1.12 Fachliche Änderungen:	29
1.1.13 Formale Änderungen:	29
5.1. Bereich Master	30
5.1.1. Auftragsnummern und freie Auftragsnummern	30
5.1.2. Buchungsarten	31
1.1.14 Abhängigkeiten	32
5.1.3. Personenkonten und Vertragspartner	33
1.1.15 Beispiel:	33
5.1.4. Belieferte Anlagen	34
5.1.5. Fraktion zu Artikel, Lagerorte	35
5.1.6. Eingang/Ausgang	37

5.1.7.	Retoure	37
5.1.8.	Garantiegeber	37
5.2.	Bereich Wiegeschein	38
5.2.1.	Sender-/Empfängerwiegeschein	38
5.2.2.	Verwiegungs-Status	38
5.2.3.	Nettokorrektur	39
5.2.4.	KFZ zu KFZ	39
5.3.	Bereich Kostenstellenanteile	41
5.3.1.	Kostenstellenfall	41
5.3.2.	Kostenstelle1	41
5.3.3.	Kostenstelle2	42
5.3.4.	Kostenstelle3	42
6	Detaillierte Ausführungen	43
6.1.	-import.xml	43
6.1.1.	Wurzel-Element <import-task>	43
6.1.2.	Element <interface>	44
6.1.3.	Element <file>	44
6.1.4.	Element <param>	44
6.2.	-interface.xml	45
6.2.1.	Wurzel-Element <interface>	46
6.2.2.	Element <file-definition>	46
6.2.3.	Element <import-schema>	46
6.2.4.	Element <import-container>	47
6.2.4.1.	Element <variable>	48
6.2.4.2.	Element <get-id-from>	48
1.1.16	Beispiel	48
6.2.4.3.	Element <filter>	49
6.2.4.4.	Element <apply-if>	50
6.2.4.5.	Element <substring>	50
6.2.4.6.	Element <regexp>	50
6.2.4.7.	Element <master>	51
6.2.4.8.	<follow-relationship-forward>:	51
1.1.17	Beispiel:	51
6.2.4.9.	Import von Master und Detail	52
6.2.4.10.	Element <multi-mapping>	52
6.3.	Multi-Mapping	53
6.3.1.	Wurzel-Element <multi-mapping>	53
6.3.2.	Element <definiton>	53
6.3.3.	Element <content>	54
6.3.4.	Besonderes	55
6.3.5.	Eindeutigkeiten	56
6.4.	-file-definition.xml	57
1.1.18	Template:	57
6.4.1.	Wurzel-Element <file-definition>	57
6.4.2.	Element <txt>	57
6.4.2.1.	Element <field>	58
6.4.3.	Element <fixed>	58
6.4.3.1.	Element <field>	58
6.4.4.	Element <xml>	58
6.4.4.1.	Element <transformer>	58
6.4.5.	Mehr zum TXT-Format	59
7	Transaktions-Management	60
7.1.	Beispiel	61
7.2.	Fehlermeldungen	61
8	Ausführungen zur Group-Funktionalität	62
8.1.	Group innerhalb einer Importdatei	62
1.1.19	Beispiel:	62

8.2. Group auf einen bereits importierten Master.....	62
1.1.20 Beispiel:	62
9 Ausführungen zum Multi - Mapping	63
10 Export	65
10.1. Einbinden Multi - Mappings im Export	65
11 Anhang	66
11.1. Abbildungsverzeichnis	66
11.2. Variablen	67
11.3. Zwischentabellen	70
11.4. Listen.....	71
11.4.1. Liste Beförderungsarten (TVW_BEFOERDERUNGSARTEN).....	71
11.4.2. Liste Retourengründe (TYF_VWRETOUREN).....	71
11.4.3. Liste Status Wiegeschein (TYF_VWSTATUS).....	71
11.4.4. Liste Buchungsarten (TYF_BUCHUNGSARTEN)	72
11.4.5. Liste Kontoklassen (TYF_KONTOKLASSEN)	72
11.4.6. Liste Kostenstellenfälle (TYF_KOSTENSTELLENFAELLE).....	72
11.4.7. Liste Kostenstellenarten (TYF_KOSTENSTELLENARTEN)	73
11.4.8. Liste Personenkonten (TST_PERSONENKONTEN).....	73
11.4.9. Liste Kostenstellen (TYF_KOSTENSTELLEN).....	73
11.4.10. Liste Artikel (TST_ARTIKEL)	73
11.4.11. Liste Lagerort (TST_LAGERORTE)	73
11.4.12. Liste Artikellagerort (TST_ARTIKELLAGERORTE).....	74
11.4.13. Liste Aufträge (TTR_AUFTRAEGE)	74
11.4.14. Liste Kreis (TYF_KFZKREISE)	74
11.4.15. Liste Land (TYF_LAENDER)	74
11.4.16. Liste Garantiegeber über Personenkonto (TST_PERSONENKONTEN)	75
11.4.17. Liste Garantiegeber über Kostenstelle (TST_KOSTENSTELLEN)	75
11.4.18. Abhängigkeit Retourkennzeichen	77
1.1.21 Eingangsverwiegungen	77
1.1.22 Ausgangsverwiegungen	77
11.4.19. Liste Artikelgruppen (TYF_ARTIKELGRUPPEN)	78
11.4.20. Liste Dialoge (TYT_DIALOGE)	79
11.4.21. Liste Buchungsarten - Kostenstellenfälle	80
11.4.22. Liste Kostenstelle1	81
11.4.23. Liste Kostenstelle2	81
11.4.24. Liste Kostenstelle3	81
11.4.25. Liste Sammelsysteme	81
11.4.26. Liste Systembetreiber	81
11.5. Formate.....	82
11.5.1. Datums- und Zeitangaben	82
11.5.2. Zahlen	82

1 Einleitung

Bei der Entwicklung von w^me.fact wurden insbesondere die hohen Anforderungen der Benutzer an Import- und Exportmöglichkeiten berücksichtigt. Hinzu kommen die Anforderungen an die Bedienung der neuen Meldeschnittstellen seitens Systembetreiber/Garantiegeber.

Zunächst ein Wort zu den Schnittstellenformaten. Zu unterscheiden sind:

- Meldung
- Transfer
- Waageschnittstelle

1.1.1 Meldung

Für die Meldung von Entsorgungs- und Verwertungsnachweisen an Systembetreiber und Garantiegeber im Rahmen des Nachweiswesens wurde diese Schnittstelle definiert. w^me.fact bedient diese Schnittstelle komplett. Als eine Sonderform des Transfers ist die Möglichkeit der Meldung an den Vertragspartner zu betrachten. Hier werden Meldungen an die Vertragspartner im Format der Transferschnittstelle bereitgestellt.

1.1.2 Transfer

Dieses Format ist für den Datenaustausch zwischen verschiedenen w^me.fact - Installationen (z.B. Anlagenbetreiber zu Vertragspartner) entwickelt worden. Es können umfangreiche Daten einschließlich der benutzerdefinierten Stammdaten ausgetauscht werden.

1.1.3 Waageschnittstelle

Der Begriff wurde in Anlehnung an das durch die DSD in 1996 verteilte Programm WME gewählt und kennzeichnet die Möglichkeit, Daten aus verschiedenen Vorsystemen frei zu importieren. Dies war ursprünglich für den Import von Wiegescheindaten vorgelagerter Waagesysteme gedacht.

Unter der Begrifflichkeit Waageschnittstelle verbirgt sich die gesamte Funktionalität des Imports / Exports von Verwiegungsdaten in beliebige .txt oder .XML-Formate. w^me.fact bietet die Funktionalität der Aufbereitung und der freien Zuordnung von Importdaten zu den Zielvariablen des w^me.fact. Analog dem Import können Daten über den gleichen Weg in spezielle Formate exportiert werden.

Bitte beachten Sie die Beschränkung der Waageschnittstelle auf Verwiegungsdaten. Sollen andere Informationen, insbesondere benutzerdefinierte Stammdaten, in w^me.fact importiert oder aus w^me.fact exportiert werden, so kann hierzu die Transferschnittstelle genutzt werden.

Ein Wort zum Schluss:

Die Waageschnittstelle wurde entwickelt mit dem Ziel, dem erfahrenen IT - Mitarbeiter die Datenaufbereitung und den Import von Daten in w^me.fact zu ermöglichen, ohne sonstige Programmierertools verwenden zu müssen. An dieser Stelle sei darauf hingewiesen, dass die zu importierenden Daten auch mit eigenen Tools für die Transferschnittstelle aufbereitet und damit direkt in w^me.fact importiert werden können.

1.1. Gegenstand dieses Dokumentes

Das vorliegende Dokument beschränkt sich auf die Beschreibung der Funktionalitäten und Definitionen der Waageschnittstelle für Modul E. Zunächst wird in den Kapiteln 1-5 eine Übersicht gegeben, bevor ab Kapitel 6 die technischen Details im Einzelnen erörtert werden. Die im Dokument angegebenen Beispiele für XML-Definitionsdateien beziehen sich auf das mitgelieferte Import-Beispiel „WsBA31Beispiel“, zu finden im Verzeichnis <wme.fact>\Schnittstelle.

1.2. Glossar

Routine	Bezeichnet die komplette Definition eines Imports/Exports
Multi-Mapping	Bezeichnet die Umschlüsselung von einem Wert auf einen anderen. Multi-Mappings werden in eigenen XML-Dateien definiert.
Schnittstellen-Definition	Bezeichnung für eine -interface.xml
Zwischentabellen	Temporäre Tabellen der ^{w^me.fact} - Datenbank, in die die Waageschnittstelle Daten importiert.

1.3. Liste der verwendeten Symbole

Symbol	Funktion	Erklärung
■	Auflistung	Auflistungen, welche keiner bestimmten Reihenfolge unterliegen
①	Aufzählung	Abläufe mit einer bestimmten Reihenfolge werden durchnummeriert dargestellt. Diese Nummerierungen finden sich auch in den Screenshots wieder. Der jeweilige Bereich auf welchen sich die Nummer bezieht, wird grün umrahmt dargestellt.
👉	Tipp	Dieses Symbol hebt wichtige Tipps und Informationen hervor
⚠	Wichtiger Hinweis	Dieses Symbol gibt Hinweise auf wichtige Informationen
→	Menü-Pfade	Bei Zugriff auf ein Menü/Untermenü gilt die Syntax: „Menü → Untermenü“

1.4. Übersicht

Für die Realisierung der Waageschnittstelle wurde ein kombinierter Ansatz aus Bedienung in der Oberfläche von wm^e.fact und der freien Konfiguration auf XML-Ebene gewählt.

Kern der Waageschnittstelle sind grundsätzlich XML-Dateien, in denen die gesamte Konfiguration hinterlegt wird:

- Definition der zu importierenden/exportierenden Daten (-file-definition.xml)
- Zuordnung der Import-/Exportdaten zu den Zielvariablen in wm^e.fact (-interface.xml)
- Steuerung aller beteiligten Dateien und Anknüpfung an wm^e.fact (-import.xml bzw. -export.xml)
- Optional besteht die Möglichkeit, Import-/Exportdaten nach einem beliebigen Muster umzuschlüsseln (Multi-Mapping)

wm^e.fact stellt diese Dateien jeweils bei der Definition einer Waageschnittstelle als Templates (standardisierte Hüllen) zur Verfügung. Die endgültige Anpassung ist anschließend direkt in den XML-Dateien vorzunehmen.

Neben der Bereitstellung der Templates wird in den Dialogen von wm^e.fact zur Waageschnittstelle die gesamte Funktionalität zur Ansteuerung des Datenimports/-exports bis hin zur Nachbearbeitung der importierten Daten und deren Übernahme in den Echtdatenbestand bereit gestellt.

Durch die flexible Auslegung der Waageschnittstelle können Daten in verschiedenen Formaten importiert bzw. exportiert werden:

- XML
- Variable Satzlänge (CSV), mit/ohne Header
- Feste Satzlänge (nur Import)

Des Weiteren müssen die zu importierenden Daten nicht zwingend dem wm^e.fact - Datenmodell entsprechen. Auch hier bietet die Waageschnittstelle die notwendige Flexibilität um Daten sowohl umzuschlüsseln (über Multi-Mapping) als auch zu bearbeiten (über Reguläre Ausdrücke).

Ein Import/Export über Direktzugriff auf andere Datenbanken ist nicht möglich. Waagesysteme müssen die in wm^e.fact zu importierenden Daten also zunächst in einem der oben genannten Formate exportieren.

Schematisch lässt sich der Importvorgang folgendermaßen darstellen (der Exportvorgang gilt analog in umgekehrter Richtung, allerdings ohne Schritt 2):

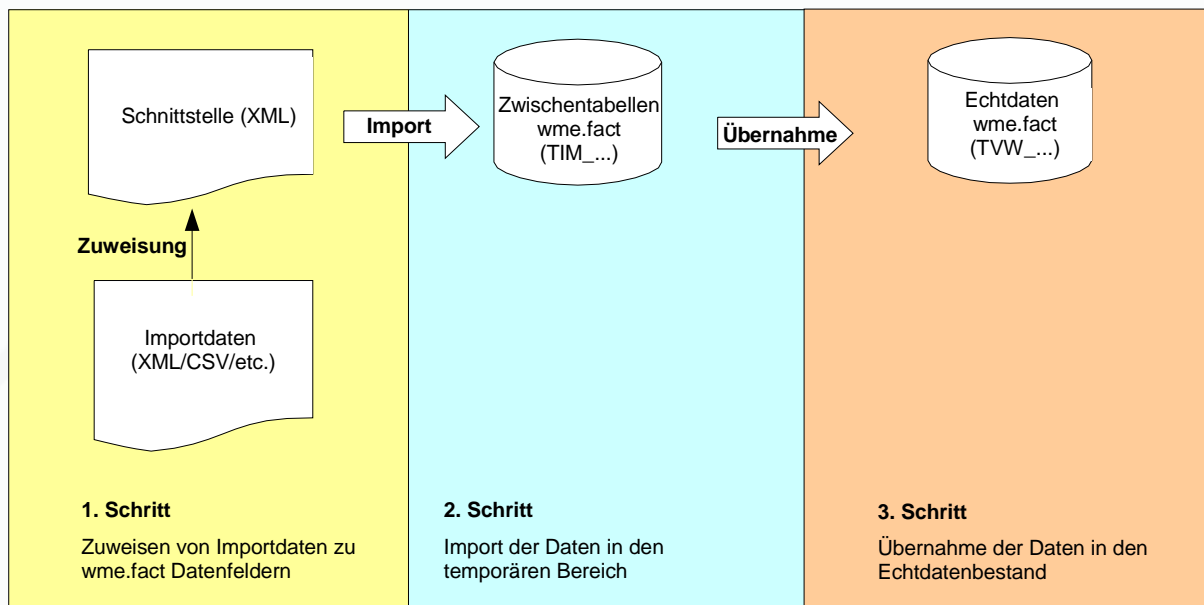


Abbildung 1-1: Schema Importvorgang

Deutlich werden die folgenden Arbeitsschritte:

- ① Zuweisen von Feldern der Importdaten-Datei zu Variablen in ^{wm}e.fact einschl. der Möglichkeit der Umschlüsselung, sowie der Konvertierung von Daten. Sämtliche Konfigurationen erfolgen über XML-Dateien.
- ② Import der Daten in „Zwischentabellen“ (Übersicht der Zwischentabellen siehe [Anhang 11.3](#)) und erste Validierungen; Bearbeitungs- sowie umfangreiche Löschfunktionen anhand von ^{wm}e.fact – Dialogen.
- ③ Übernahme der importierten Daten in den Echtdatenbestand, umfassende letzte Validierungen, bekannte Bearbeitungsfunktionen über die Dialoge.

Der Rahmen (die Routine) für einen solchen Vorgang wird im Programm unter **Transfer → Waage Schnittstelle → Daten importieren** angelegt und verwaltet.

Nachfolgend werden zunächst in Kurzform ein Importvorgang sowie ein Exportvorgang durchgängig beschrieben, bevor in den dann folgenden Kapiteln auf die Besonderheiten und Möglichkeiten eingegangen wird.

2 Import

2.1. Zusammenhänge

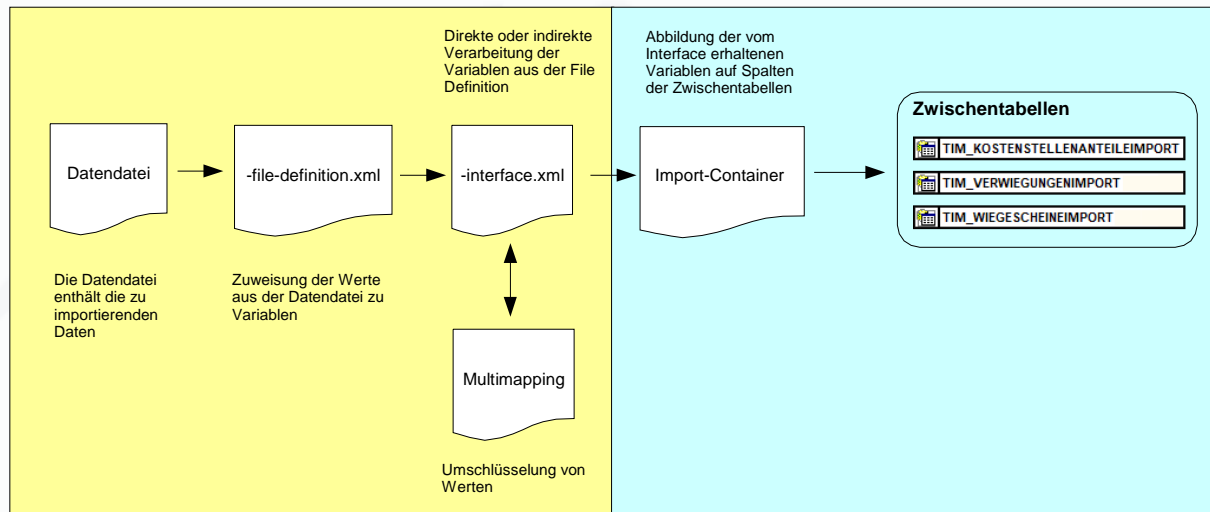


Abbildung 2-1: Schema XML - Dateien, Zwischentabellen

Die Steuerung des Importvorganges, also die Zuordnung von Daten aus der zu importierenden Datei zu Spalten in den temporären Tabellen, geschieht über die XML-Dateien der Routine (gelber Bereich). Dies sind die Dateien, die zu konfigurieren sind.

Im Dialog von w^me.fact werden die Steuerungsinformationen der Import-Routine (-import.xml) dynamisch während der Laufzeit ausgelesen. Die Importroutine ist damit der Einstiegspunkt des gesamten Importvorganges (im Dialog w^me.fact „Routine“ genannt). In ihr werden die Schnittstellendefinitionen (-interface.xml) sowie die zu importierenden Datendateien angegeben.

Die Schnittstellendefinition (-interface.xml) ist der „Vermittler“ von Quell- zu Zielspalten. Einer Schnittstellendefinition kann genau eine File-Definition (-file-definition.xml) und beliebig viele Multi-Mappings zugeordnet werden.

In der File-Definition (-file-definition.xml) wird das Format der zu importierenden Datendatei festgelegt und im Falle von CSV oder fester Satzlänge werden die Spalten der zu importierenden Datei definiert und benannt.

In den Multi-Mapping-Dateien (diese XML-Dateien werden nicht vom Dialog erstellt) können Umschlüsselungen von Importdaten hinterlegt werden.

Beispiel:

Die Daten der alten WME BA 31/32-Schnittstelle enthalten in der Spalte: „STO_GRD“ die Inhalte „00“ für „keine Retoure“ oder „20“ für „Retoure“. Diese können mit Hilfe der Multi-Mappings auf die neuen Kennzahlen für Retourkennzeichen und Retourengrund umgeschlüsselt werden.

2.2. Grober Ablauf

Zur Durchführung eines Imports sind die folgenden, grundlegenden Schritte notwendig (einen detaillierteren Ablauf finden Sie unter [2.3. Fahrplan Import](#)):

1. Anlegen einer Routine über den Dialog Transfer → Waage Schnittstelle → Daten importieren
2. Bearbeitung der vom Dialog erstellten XML-Dateien
3. Import der Daten

Die Schritte 1 und 2 werden in der Regel einmalig durchzuführen sein. Schritt 1 sollten Sie vom Dialog ausführen lassen, da die erstellten XML-Dateien bereits sehr viele Pflichtfelder mit den korrekten Werten enthalten.

2.2.1. Anlegen einer Routine

Die Verwaltung der Import-Schnittstellen erfolgt über den Dialog **Transfer → Waage Schnittstelle → Daten importieren**. In diesem Dialog wird eine Schnittstelle initial angelegt (und standardisierte Templates für die benötigten XML - Dateien erstellt) und später der Import ausgeführt.

The screenshot shows the 'Waage Schnittstelle Import' dialog box. At the top, the 'Routine' field is set to 'WsBA31Beispiel'. Below this, there are two buttons: 'Erstellen' and 'Importieren'. The main area of the dialog is divided into two tabs: 'Schnittstellen' and 'Logfile'. The 'Schnittstellen' tab is active, and within it, there are two sub-tabs: 'Tabelle' and 'Datensatz'. The 'Datensatz' sub-tab is selected, showing the following fields:

- Bezeichnung:** BA31
- Bereich:** Verwiegunen (komplett)
- Dateiname:** data/BEISPIEL_BA31.321
- Format:** Feste Satzlänge
- Separator:** (empty field)

At the bottom of the 'Datensatz' section, there is a button labeled 'Datei auswählen'.

Abbildung 2-2: Dialog Waage Schnittstelle Import (Erstellen)

1.1.4 Felddesreibungen

Master „Waage Schnittstelle Import“


Feld: Routine	Eingabe eines freien Namens für die Routine
Button: Erstellen	Nach der Zuordnung min. eines Details führt die Betätigung des Buttons zur Erstellung der XML-Dateien (Templates)
Button: Importieren	Die Betätigung löst den Import der in den Details zugewiesenen Dateien (siehe Detail Schnittstellen: Dateiname) aus.

Detail „Schnittstellen“

Feld: Bezeichnung	Eingabe einer Bezeichnung für die in der Routine verwendeten Schnittstellen (mehrere sind möglich)
Feld: Bereich	Es muss der Bereich ausgewählt werden, in den importiert werden soll. Der Benutzer kann auswählen, welche Daten die zu importierende Datei enthält: <ul style="list-style-type: none"> - Daten für den Verwiegungsmaster, die Details Wiegescheine und die Kostenstellen „Verwiegung (komplett)“ - nur Verwiegungsmaster „Verwiegung (nur Master)“ - nur Wiegescheindetails „Verwiegung (nur Wiegescheine)“ - nur Kostenstellendetails „Verwiegung (nur Kostenstellenanteile)“. Entsprechend der Auswahl werden die XML - Templates erstellt.
Feld: Dateiname	Angabe des Namens der zu importierenden Datendatei; Auswahl über einen Dialog durch Betätigen des Buttons „Datei auswählen“
Feld: Format	Auswahl des Formats der Importdatei; derzeit können XML - Dateien, Textdateien mit fester Satzlänge oder mit variabler Satzlänge importiert werden.
Feld: Separator	Angabe des Feldtrenners bei Dateien mit variabler Satzlänge. Sind die Felder mit einem Tabulator getrennt, ist die Angabe „TAB“ notwendig. Ansonsten werden die Feldtrenner gemäß ihrer Darstellung in der Datei angegeben (z.B.: „“)






Detail „Logfile“

Feld: Datum	Datum, an dem mit dieser Routine ein Import gestartet wurde
Feld: Quelle	Name der importierten Waagedatei
Feld: Protokolldatei	Name der Protokolldatei des Imports im Verzeichnis <wme.fact>Protokolleimport; enthält eine Liste der importierten Datensätze
Feld: Fehlerdatei	Name der Fehlerdatei des Imports im Verzeichnis <wme.fact>Protokolleimport; enthält soweit als möglich eine Beschreibung der abgewiesenen Sätze
Feld: Anzahl Importiert	Anzahl der importierten Sätze
Feld: Anzahl Fehler	Anzahl der abgewiesenen Sätze

 Weitere möglicherweise vorhandene Fehlerhinweise des Imports befinden sich im Verzeichnis <wme.fact>\logging

Sind alle Informationen angegeben und erfolgreich gespeichert, so wird mit Betätigen des Buttons „Erstellen“ im Verzeichnis <wme.fact>\Schnittstelle ein Unterverzeichnis mit dem Namen der Routine (hier: WsBA31Beispiel) und innerhalb des Verzeichnisses eine WsBA31Beispiel-import.xml angelegt. Der im Feld „Bezeichnung“ hinterlegte Eintrag bestimmt den Dateinamen der erzeugten -file-definition.xml und der -interface.xml („BA31-file-definition.xml“ und „BA31-interface.xml“).

Die so erzeugten XML-Dateien enthalten das Grundgerüst für einen Import, d.h. alle benötigten XML-Dateien werden als Templates bereitgestellt und mit den im Dialog gemachten Angaben vorbelegt. Änderungen bzw. die endgültige Ausformulierung der Schnittstelle sind anschließend mit einfachen Texteditoren (z.B. Notepad) oder XML-Editoren vorzunehmen.

-  Wird eine Routine über den Dialog gelöscht, so wird auch konsequenterweise das entsprechende Verzeichnis mit den darin befindlichen Dateien gelöscht.
Löschen Sie Routinen nicht manuell aus dem Verzeichnis <wme.fact>\Schnittstelle, sondern nur über den Dialog, damit der Löschvorgang auch in der Datenbank korrekt ausgeführt werden kann.
-  Definitionen von Routinen und Schnittstellen gelten anlagen- und modulübergreifend; innerhalb der Installation kann von jeder Anlage auf die einmal definierte Routine zugegriffen werden
-  In der Datenbank selbst wird der Name der Routine sowie die erfassten Schnittstelleninformationen mit der Kennzeichnung hinterlegt, ob es sich um einen Import oder Exportvorgang handelt (hier: WSBA31Beispiel als Import).
Alle im Detail erfassten Informationen werden zudem in den entsprechenden Templates (XML-Dateien) hinterlegt und von dort jeweils beim Importvorgang aktuell ausgelesen. Um diese Templates zu finden, wird nach der Datei <Name der Routine>-import.xml im Unterverzeichnis <wme.fact>\Schnittstelle\<Name der Routine>\ gesucht, die wiederum alle anderen Pfadinformationen enthält.
-  Sollen Importroutinen von einer w^me.fact – Installation zu einer nächsten transferiert werden, so reicht es aus, das Verzeichnis der jeweiligen Routine aus <wme.fact>\Schnittstelle in das Zielverzeichnis <wme.fact>\Schnittstelle zu kopieren. Beim nächsten Öffnen des Import-Dialoges wird die Routine automatisch erkannt.
-  Legen Sie eigene Unterverzeichnisse für die Schnittstellen- und Multi-Mapping-Definitionen an und referenzieren Sie von der -import.xml auf diese Dateien. Änderungen im Dialog werden sich dann nicht mehr auf diese „ausgelagerten“ XML-Dateien auswirken.

2.2.2. Konfiguration der XML - Dateien

Wie bereits erwähnt, werden die folgenden XML - Dateien vom Dialog initial erstellt:

- -import.xml
- -file-definition.xml
- -interface.xml

Evtl. nötige Multi-Mapping XML-Dateien müssen manuell erstellt werden.

- import.xml

Die Konfiguration dieser Datei kann in der Regel vollständig über den Dialog erfolgen.

- file-definition.xml

Dies ist die erste Datei die konfiguriert werden sollte. Im Fall von fester oder variabler Satzlänge ohne Überschriften werden die Feldnamen definiert, auf die die -interface.xml dann zugreifen kann. Im Fall von nicht schemakonformem XML-Import kann ein Transformationsszenario angegeben werden.

- interface.xml

Hier werden die Importdaten den Zielvariablen des Systems zugewiesen, entweder direkt oder über Multi-Mapping, bzw. über Reguläre Ausdrücke.

Die folgenden Ausführungen beziehen sich auf das mitgelieferte Beispiel *WsBA31Beispiel*. Dieses Beispiel importiert Daten, die im Format der Spezifikation der alten WME - Schnittstelle (siehe Dokument „Schnittstellenbeschreibung zu WME - Neu“; Bestell - Nr. 205.D4; Dezember 1997; Abschnitt „B.1 Input-/Output-Daten“) vorliegen.

- ⚠ Um das Beispiel ausführen zu können, muss zunächst ein Lagerort und mindestens einen Artikel in den w^me.fact - Stammdaten angelegt werden:
Lagerort: Lagerort = Halle1
Artikel: Artikelgruppe = 0600, Artikelnummer = 0600, gültig ab = 01.01.2004, Lagerort = Halle1
- ⚠ Das Beispiel ist auf die mitgelieferten Daten abgestimmt. Sollen eigene BA31 - Daten importiert werden, so muss gegebenenfalls die Konfigurationen in den XML-Dateien angepasst werden.


2.2.2.1. Beschreibung der -import.xml (WsBA31Beispiel-import.xml)

```
1 <?xml version="1.0" encoding="ISO-8859-1"?>
2
3 <import-task
4   xmlns="http://www.dsd-ag.de/wme-fact/transfer/task"
5   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
6   name="WsBA31Beispiel"
7   transaction="true"
8   xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/task ../../Config/Transfer/schema/task.xsd">
9
10  <interface href="BA31-interface.xml">
11    <file href="data/BEISPIEL_BA31.321"/>
12  </interface>
13 </import-task>
```

Abbildung 2-3: WsBA31Beispiel-import.xml

Die Datei WsBA31Beispiel-import.xml referenziert im Wurzel-Element „import-task“ auf zugrunde liegenden Schemata und Namespaces.

Im Element „interface“ wird die zu verwendende Definition (BA31-interface.xml) angegeben und mit der zu importierenden Datei verknüpft.

 Die -interface.xml wird standardmäßig im selben Verzeichnis wie die -import.xml angelegt. Selbstverständlich kann aber auch hier manuell ein entsprechender Pfad hinterlegt werden.

2.2.2.2. Beschreibung der -file-definition.xml (BA31-file-definition.xml)

Die -file-definition.xml beschreibt das Datenformat und die Felder der Datendatei. Für den Fall, dass die Datendatei keine Spaltenüberschriften enthält, kann man hier Feldnamen für die Spalten einführen, um mit diesen Feldnamen in der Schnittstellen-Definition (-interface.xml) zu arbeiten.

Eine File-Definition kann i. d. R. so gestaltet werden, dass sie sowohl für den Import, als auch den Export dient.

Im Beispiel werden Daten der alten WME - Schnittstelle (BA31) importiert. Diese liegen als TXT-file mit fester Satzlänge vor. Die File-Definition dazu sieht wie folgt aus (Ausschnitt):

```

1  <?xml version="1.0" encoding="ISO-8859-1"?>
2
3  <file-definition
4      xmlns="http://www.dsd-ag.de/wme-fact/transfer/file-definition"
5      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
6      name="Format-Definition Schnittstelle BA31"
7      format="fixed"
8      xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/file-definition ../../Config/Transfer/schema/file-definition.xsd">
9
10     <!--
11     Die Positionsangaben in der Schnittstellenspezifikation für Input-/Output-Daten basieren auf 1, deshalb
12     wird hier der offset-zero entsprechend gesetzt, damit die im folgenden verwendeten Angaben für offset den
13     Zahlen der Spezifikation entsprechen
14     -->
15     <fixed offset-zero="1">
16
17     <!--
18     Mit dem Element <field> kann jedem Feld der Eingabe-Datei ein logischer Feldname zugewiesen werden.
19     Dieser Feldname steht somit in der Schnittstellen-Definiton bei den Variablenzuweisungen zur Verfügung
20     -->
21     <field name="BA_NR" length="2" offset="1"/>
22     <field name="VG_NR" length="5" offset="3"/>
23     <field name="DSD" length="11" offset="8"/>
24     <field name="KDNR_DATLI" length="6" offset="19"/>
25     <field name="MONAT_JAHR" length="6" offset="25"/>
26     <field name="FRAKT_NR" length="4" offset="31"/>
27     <field name="ANL_NR_BEL" length="6" offset="35"/>
28     <field name="ANL_NR_HER" length="6" offset="41"/>
29     <field name="DSD1" length="1" offset="47"/>

```

Abbildung 2-4: BA31-file-definition.xml (Ausschnitt)

Auch hier referenziert das Wurzel-Element („file-definition“) auf zugrunde liegende Schemata, und gibt an, in welchem Format die Importdaten vorliegen (hier: format = „fixed“).

Mit dem Element „field“ werden (frei wählbare) Feldnamen definiert und angegeben, an welcher Position in der Datendatei die jeweiligen Werte stehen. Die Datei -interface.xml wird auf diese Feldnamen zugreifen um die Werte auszulesen und zu verarbeiten.

Beispiel:

<field name = „BA_NR“.....> definiert, dass der Wert für die Belegart in der Importdatei 2 Zeichen lang ist (length = „2“) und an der ersten Stelle beginnt (offset = „1“).

Zu beachten ist, dass sich die Zuordnungssyntax in Abhängigkeit vom Format der Importdaten ändern kann. Liegen die Daten im .txt-Format mit fester Satzlänge vor, so gilt die o.g. Syntax.

Liegt eine variable Satzlänge ohne Überschriften vor und die Belegart stünde in der ersten Spalte, so würde gelten <field name = „BA_NR“ column = „1“/>

Bei variabler Satzlänge mit Überschriften würde gelten <field name = „BA_NR“ column = „Überschrift der Spalte Belegart“/>

Bei XML-Import wiederum können keine Feldnamen definiert werden, da die -interface.xml in diesem Fall direkt auf die XML-Elemente zugreift.

2.2.2.3. Beschreibung der -interface.xml (BA31-interface.xml)

Die Schnittstellendefinition (-interface.xml) stellt die Verbindung zwischen Feldnamen der File-Definition (-file-definition.xml) und Feldnamen der Datenbanktabellen her. Nachfolgend dargestellt ist ein Ausschnitt aus dem mitgelieferten Beispiel WsBA31Beispiel:

```

1  <?xml version="1.0" encoding="ISO-8859-1"?>
2
3  <interface
4    xmlns="http://www.dsd-ag.de/wme-fact/transfer/interface"
5    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
6    name="BA31"
7    container="Wareneingang"
8    xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/interface ../../Config/Transfer/schema/interface.xsd"
9    stop-on-error="false">
10
11   <file-definition href="BA31-file-definition.xml"/>
12
13   <import-schema>
14
15     <!-- Verriegelungsmaster -->
16     <import-container ref="TIM_VERWIEGUNG">
17
18       <!-- Keine Auftragsnummer in den Daten, deshalb Auftragsnummer = Wiegescheinnummer -->
19       <variable ref="AUFTRAGSNUMMER" field="WIEGE_NR"/>
20
21       <variable ref="BELEGDATUM" field="WIEGE_DAT" format="ddMMyyyy"/>
22
23       <variable ref="EINGANG" multi-mapping="buchungsart-alias/WARENEINGANG"/>
24
25       <variable ref="BUCHUNGSART">
26         <get-id-from container="BUCHUNGSARTEN">
27           <variable ref="n_Kennnummer" multi-mapping="buchungsart-alias/BUCHUNGSART_KENNR"/>
28         </get-id-from>
29       </variable>

```

Abbildung 2-5: BA31-interface.xml (Ausschnitt)


Im Element „file-definition“ wird zunächst angegeben, mit welcher -file-definition.xml diese Schnittstellendefinition arbeitet. Es kann nur genau eine -file-definition.xml angegeben werden.

Mit dem Element „variable“ werden die Feldnamen aus der -file-definition.xml mit den Datenbankfeldern verknüpft. Im einfachsten Fall, z.B. der Auftragsnummer, wird der Inhalt eines Feldes aus der file-definition direkt einem Datenbankfeld zugewiesen.

In anderen Fällen, z.B. Eingang, ist dies nicht direkt möglich, da der alte Wert für BA_NR erst umgeschlüsselt werden muss (aus 31 = Input wird umgeschlüsselt auf T = True).

Wiederum andere Variablen, z.B. der BUCHUNGSART, müssen mit einem technischen Ident (id) eines Datensatzes (z.B. id der Buchungsart) gefüllt werden, dieser steht in der Regel aber nicht direkt zur Verfügung, sondern wie im Beispiel vorgestellt nur die Kennnummer (z.B. 100 = EV aus Haushaltssammlung) der Buchungsart. Mithilfe des Elementes „get-id-from“ kann die Schnittstelle die id aber ermitteln, indem der Datensatz über die Kennnummer selektiert und die id dieses Datensatzes dann der Variablen zugewiesen wird.

Im Template einer -interface.xml werden automatisch alle Zielvariablen angegeben, für die ein Wert importiert werden kann. Können einzelne Variablen nicht über Datenfelder der zu importierenden Datei gefüllt werden, so können diese Felder auskommentiert werden (<!-- ... -->). Die Bedeutung / Validierung der einzelnen Variablen, insbesondere deren Verknüpfung zu den Feldern in der Oberfläche von wme.fact, findet sich im Anhang.

 Nach der Erstellung einer neuen Importroutine finden Sie in der -interface.xml Einträge der Form:

```
<variable ref="AUFTRAGSNUMMER" field="__hier_den_Namen_der_Spalte_eintragen__"/>
```

Für „__hier_den_Namen_der_Spalte_eintragen__“ müssen Sie z.B. den Feldnamen eintragen, den Sie in der -file-definition.xml definiert haben. Die Einzelheiten zu den Variablen, fachliche Bedeutung, Format, etc. sind im Kapitel 5 beschrieben.

2.2.2.4. Verknüpfen von -interface.xml und -file-definition.xml

In unserem Beispiel ist die BA31-interface.xml bereits auf die BA31-file-definition.xml referenziert. Die Verbindung der Felder erfolgt in der Form:

```
<variable ref = <zielvariable> field = <quellvariable> format = „ddMMyyyy“>
```

Ausschnitte aus der BA31-interface.xml:

```
<variable ref="BRUTTO" field="BRUTTO">  
<variable ref="BRUTTOZEIT" field="BRUTTOZEIT" format="HH:mm">  
<variable ref="TARA" field="TARA">  
<variable ref="TARAZEIT" field="TARA_ZEIT" format="HH:mm">  
<variable ref="NETTOKORREKTUR" default="0">
```

Zusätzlich können der Variablen weitere Eigenschaften hinzugefügt werden, wie z B.:

format

Gibt an, in welchem Format ein Wert vorliegt, so dass er korrekt interpretiert werden kann. Eine Übersicht möglicher Formatangaben finden Sie im [Anhang 11.5](#).

default

Vorbelegung einer Variablen mit einem festen Wert

Weitere Möglichkeiten (wie z.B. Gruppierungen, Parametrisierung, Bearbeitung über Reguläre Ausdrücke, etc.) werden in [Kapitel 6.2](#) behandelt.

2.2.2.5. Multi - Mapping (Schlüsseldefinitionen)

Können die Importdaten nicht direkt einer Zielvariablen zugewiesen werden, sondern müssen in einer einfachen Form transformiert werden, so kann mit Hilfe von Multi-Mappings eine Umschlüsselung vorgenommen werden.

Beispiel:

Im Eingangs dargestellten Beispiel sollen die Retourkennzeichen der alten WME - Schnittstelle im Feld RETOUR_KZ mit den Werten „T“ oder „F“ auf die Einträge Retouren in wme.fact – Kennnummer „1“ für „[keine Retoure]“ und Kennnummer „2“ für „Nicht spezifikationsgerecht“ (TYF_VWRETOURKENNZEICHEN über interface.xml) - umgeschlüsselt werden.

Um ein Multi - Mapping zu definieren wird zunächst aus dem Schema „multi-mapping.xsd“ (zu finden im Verzeichnis <wme.fact>\Config\Transfer\schema) eine entsprechende .xml abgeleitet (im Beispiel retour-mm.xml). Quell- und Zielvariablen werden definiert und mit Werten gefüllt.

```


1  <?xml version="1.0" encoding="ISO-8859-1"?>
2
3  <multi-mapping
4    xmlns="http://www.dsd-ag.de/wme-fact/transfer/multi-mapping"
5    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
6    xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/multi-mapping ../../Config/Transfer/schema/multi-mapping.xsd"
7    name="retoure-mm">
8
9    <definition>
10     <source name="RETOUR_KZ_ALT"/>
11     <target name="RETOURE_KENNR"/>
12   </definition>
13
14   <content>
15
16     <record>
17       <source name="RETOUR_KZ_ALT">F</source>
18       <target name="RETOURE_KENNR">1</target>
19     </record>
20
21     <record>
22       <source name="RETOUR_KZ_ALT">T</source>
23       <target name="RETOURE_KENNR">2</target>
24     </record>
25
26     <default>
27       <target name="RETOURE_KENNR">1</target>
28     </default>
29   </content>
30
31 </multi-mapping>

```

Abbildung 2-6: Multi-Mapping retour-mm.xml

Der Definitions-Teil, Element *definition*, deklariert die *Quell*-Felder (=source) und die *Ziel*-Felder (=target). Es sind jeweils beliebig viele Felder möglich.

Im Content-Teil ist der eigentliche Inhalt enthalten, also die Abbildungssätze (Element *record*). Das default - Element ist optional. Es definiert das Ergebnis, welches genommen wird, wenn keiner der vorherigen Sätze zutrifft. Im vorliegenden Beispiel soll also immer dann, wenn eine Retourvariable nicht aufgelöst werden kann, der Wert „1“ für „keine Retoure“ gesetzt werden.

 Es können mehrere Source-Felder bzw. Target-Felder verknüpft werden. Verknüpfungsbedingung ist immer <and>.

 Wählen Sie die Namen der Quell- und Zielfelder so, dass Sie am Namen erkennen, ob es sich um ein Quell- oder Zielfeld handelt.

2.2.2.6. Einbinden von Multi - Mappings in interface.xml

Eine Schlüsseldefinition wird an der Stelle herangezogen, an welcher die neuen Werte benötigt werden, also in der Schnittstellendefinition bei der Belegung der Variablen. Zunächst ist es notwendig, die Quell-Felder mit den Feldern der Importdatei zu verknüpfen. Dies geschieht in eigenen *multi-mapping*-Elementen, welche als letzte Elemente einer Schnittstellendefinition möglich sind.

Syntax:

```
<multi-mapping name="retoure-alias" href="Multimapping\retoure-mm.xml">
  <source ref="RETOUR_KZ_ALT" field="RETOUR_KZ"/>
</multi-mapping>
```

Da innerhalb einer Schnittstelle beliebig viele Schlüsseldefinitionen herangezogen werden dürfen, ist es notwendig einen Alias mit dem Attribut *name* (hier: *retoure-alias*) zu vergeben. Mit *href* verweist man auf die eigentliche Datei, die in diesem Beispiel in einem Unterverzeichnis \Multimapping abgelegt ist. Dann wird das Quell-Feld (hier: RETOUR_KZ_ALT) aus der Mapping - Datei mit der Spalte der Importdatei (hier: RETOUR_KZ) verknüpft.

Die Referenz auf das *Ziel-Feld* (hier: RETOURE_KENNR) erfolgt bei der Belegung der Variablen mit:

Beispiel:

```
<variable ref= »RETOURE »>
  <get-id-from container="RETOUREN">
<variable ref="n_Kennnummer" multi-mapping="retoure-alias/RETOURE_KENNR"/>
  </get-id-from>
</variable>
```

Der Wert des Attributs *multi-mapping* setzt sich aus zwei Teilen zusammen:

- Dem Alias der Schlüsseldefinition: „retoure-alias“
- Dem Ziel-Feld aus „retoure-alias“: „RETOURE_KENNR“

⚠ Beim Export wie auch beim Import können dieselben Schlüsseldefinitionen herangezogen werden. Wird beim Import bspw. „RETOUR_KZ_ALT“ auf „RETOUR_KZ“ abgebildet, so wird beim Export „RETOUR_KZ“ auf „RETOUR_KZ_ALT“ umgekehrt abgebildet ([Beispiel](#))

```
<multi-mapping name="retoure-alias" href="Multimapping\retoure-mm.xml">
  <target ref="RETOUR_KZ" field="RETOUR_KZ_ALT"/>
</multi-mapping>
```

Einschränkungen des Multi-Mappings:

⚠ Es wird für die Verwendung des Multi-Mappings im Import jedem Source-Wert eindeutig ein Target- Wert zugeordnet (1:n).

Eine Multi-Mapping-Datei kann sowohl im Export als auch im Import Verwendung finden. In diesem Falle müssen die Werte Source/Target in einer 1:1-Beziehung stehen.

2.2.3. Import der Daten

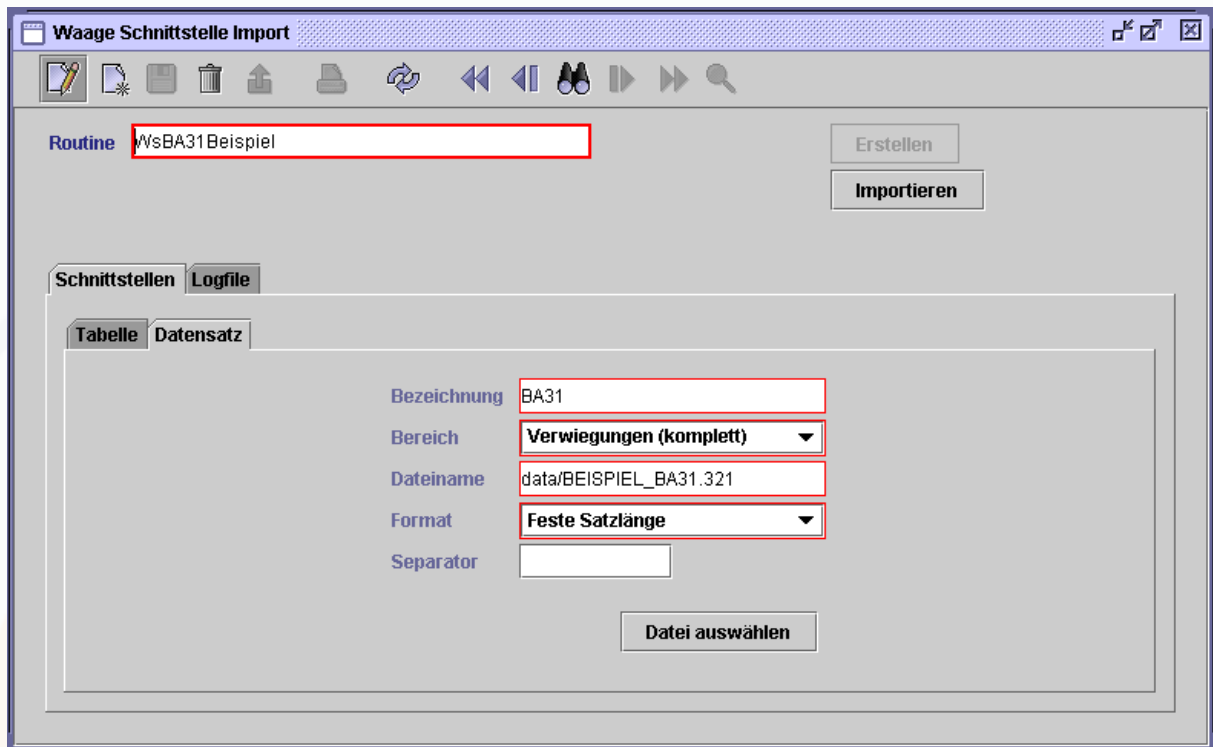


Abbildung 2-7: Dialog Waage Schnittstelle Import (Importieren)

Nachdem die Konfiguration der XML-Dateien abgeschlossen ist, können Daten importiert werden. Dazu wird der Import-Dialog erneut aufgerufen (**Transfer** → **Waage Schnittstelle** → **Daten importieren**), und die erstellte Routine ausgewählt. Mit betätigen des Buttons „Importieren“ werden die Daten aus der Importdatei über die Schnittstelle eingelesen und in den Zwischentabellen abgespeichert. Dort können die Daten, wie in [Kapitel 4](#) beschrieben, weiterverarbeitet werden.

- ⚠ Das Detail der Schnittstelle kann auf Bearbeiten gesetzt und der Name der zu importierenden Datei geändert und gespeichert werden. Eine einmal konfigurierte Schnittstelle kann somit für den Import von unterschiedlichen Datendateien verwendet werden.
- ⚠ Wird der Bereich geändert, so werden die XML - Templates neu erstellt (Sicherheitsabfrage). Das bedeutet, dass die bestehenden und gegebenenfalls bereits geänderten XML-Dateien bei diesem Vorgang überschrieben werden. Es sollte daher vorher eine Sicherung der Dateien vorgenommen werden.

2.3. Fahrplan Import

Nachdem nun ein erster Einstieg in die Importfunktionalitäten ^{wm}e.fact geschafft ist, soll nachfolgend der gesamte Ablauf ergänzend noch einmal dargestellt werden:

1.1.5 Oberfläche ^{wm}e.fact

- ① Anlegen einer Routine in der Oberfläche von ^{wm}e.fact (**Transfer → Waage Schnittstelle → Daten importieren**)
- ② Innerhalb der Routine werden ein oder mehrere Bereiche angegeben, in den Daten importiert werden sollen (komplette Verriegung oder nur Wiegescheine etc.)
- ③ Mit betätigen des Button „Erstellen“ werden die entsprechenden XML-Templates im Verzeichnis <wme.fact>\Schnittstelle\<Name der Routine> angelegt

1.1.6 XML-Editor

- ④ Im ersten Schritt ist die -file-definition.xml anzupassen, also die Definition der zu importierenden Daten vorzunehmen
- ⑤ Anschließend werden die über die -file-definition.xml deklarierten Import-Variablen, soweit möglich, direkt den ^{wm}e.fact - Zielvariablen in der -interface.xml zugeordnet
- ⑥ Können Werte der Importdatei nicht direkt Zielvariablen zugeordnet werden, da sie in anderer Bedeutung vorliegen (siehe im Beispiel: BA_NR), so können anhand von Umschlüsselungstabellen (Multi-Mappings) entsprechende Datenaufbereitungen vorgenommen werden.

1.1.7 Oberfläche ^{wm}e.fact

- ⑦ Erneuter Aufruf der definierten Schnittstelle (**Transfer → Waage Schnittstelle → Daten importieren**) und starten des Importvorgangs (Button „Importieren“)
- ⑧ Kontrolle und Nachbearbeitung der importierten Daten (**Transfer → Waage Schnittstelle → Daten bearbeiten** oder **Transfer → Waage Schnittstelle → Daten löschen**)
- ⑨ Abschließend können die importierten Daten in den Echtdatenbestand übernommen werden (**Transfer → Waage Schnittstelle → Daten übernehmen**)

3 Export

Der Dialog und Tabellenaufbau sind analog zum Import gestaltet. Vergleichbare Funktionalitäten werden daher nicht mehr explizit beschrieben.

3.1. Allgemein

Der Datenexport basiert auf folgende Konzepte:

- Konfigurierbarkeit des Exportes mit Hilfe von XML
- Export in verschiedene Formate (.txt, .xml)
- Auslösen des Exports über den Dialog

Die XML-Konfiguration beschreibt das Ausgabe-Format und die Abbildung der Attribute auf Ausgabe-Spalten bzw. XML-Elemente. Es ist ebenfalls definiert, welche Entität exportiert wird, jedoch nicht die Filterung der Daten.

3.2. Anlegen einer Export-Task

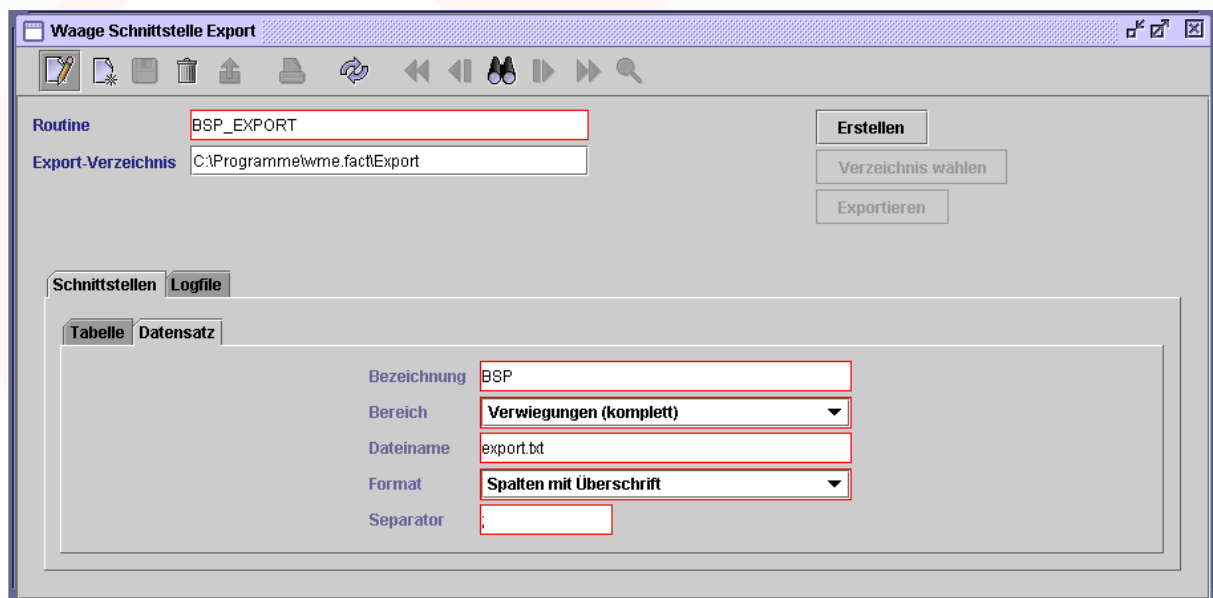


Abbildung 3-1: Dialog Waage Schnittstelle Export

Analog dem Import ist beim Export zunächst eine Routine anzulegen (zu Bedienung, Tabellenstruktur etc. siehe [Kapitel 2.2.1](#))

1.1.8 Besonderheiten:

- ⚠ Ein Export kann nicht im Format „feste Satzlänge“ erfolgen.
 - ⚠ Ein Export im Format der variablen Satzlänge (Spalten mit oder ohne Überschrift) ist beschränkt auf die Bereiche
 - nur Verwiegungsmaster „Verwiegunung (nur Master)“
 - nur Wiegescheindetails „Verwiegunung (nur Wiegescheine)“
 - nur Kostenstellendetails „Verwiegunung (nur Kostenstellenanteile)“,
- Da sich die Abhängigkeiten der Entitäten für den Bereich „Warenein-/ausgang“ nicht sinnvoll mit der flachen Struktur dieses Formates abbilden lassen.
- ⚠ Ein Export im XML-Format ist für alle Bereiche ohne Einschränkung möglich.

Die import-, interface-, file-definition und das Multi-Mapping finden analog der Importbeschreibung, mit folgenden Abweichungen, Anwendung:

```

1  <?xml version="1.0" encoding="ISO-8859-1"?>
2
3  <file-definition xmlns="http://www.dsd-ag.de/wme-fact/transfer/file-definition"
4     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5     name="Format-Definition Schnittstelle BSP"
6     format="txt"
7     xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/file-definition
8                          file:/C:/Programme/wme.fact/Config/Transfer/schema/file-definition.xsd">
9
10     <txt separator=";" header="true"/>
11
12 </file-definition>

```

Abbildung 3-2: -file-definition.xml (Export)

Die -file-definition.xml für den Export enthält im Datenteil nur noch den Hinweis auf das zu verwendende Format (hier: txt), den angegebenen Separator (hier: ;) und die Angabe, ob Spaltenüberschriften ausgegeben werden sollen (Header: true). Die Variablendefinition der Zieldatei erfolgt direkt bei der Variablenzuweisung in der -interface.xml des Exports.

- ⚠ Für die Verwendung des Multi-Mappings im Export muss jedem Target-Wert eindeutig ein Source-Wert zugeordnet werden (1:n).

4 Nachbearbeitungsdialoge für den Import

4.1. Dialog: Waage Schnittstelle Pflege

In dem Dialog des Menüpunktes **Transfer → Waage Schnittstelle → Daten bearbeiten** können die Daten der Zwischentabelle nach dem Import bearbeitet werden. Validierungen finden nur gegen die Referenztabellen statt (z.B. Buchungsarten), jedoch nicht auf fachliche Korrektheit sowie gegen die Daten des Echtbestandes (doppelte Wiegescheine etc.)

Die Eindeutigkeit eines Wiegescheins ist hierbei wie folgt definiert:

„An einer Anlage darf eine Wiegescheinnummer zu einem Wiegedatum nur einmal existieren.“

Abbildung 4-1: Dialog Waage Schnittstelle Pflege

4.2. Dialog: Waage Schnittstelle Löschen

In dem Dialog des Menüpunktes **Transfer → Waage Schnittstelle → Daten löschen** können importierte Sätze in den Zwischentabellen nach Art der Verwiegung gesamt oder nach Fehlerklassen unterschieden gelöscht werden. Zudem können einzelne Importvorgänge anhand des Importdatums gelöscht werden.

Abbildung 4-2: Dialog Waage Schnittstelle Löschen

Der Button „Anzahl ermitteln“ führt eine Abfrage auf die Zwischentabellen aus und zeigt die Anzahl der vorhandenen Sätze, unterschieden nach deren Fehlerstatus, an. Über die Checkboxes im oberen Teil des Dialogs können Teilbereiche zur Löschung vorgesehen werden. Der Löschvorgang wird mit Betätigen des Buttons „Löschen“ ausgeführt. Mit dem Löschvorgang werden die Daten physikalisch aus den Zwischentabellen entfernt.

Über die Combobox „Importdatum“ können einzelne Importvorgänge, nach Datum / Uhrzeit des Importvorgangs getrennt, betrachtet werden.

Die Informationen im Detail „Anzahl“ lesen sich wie folgt:

1.1.9 Bereich \ Anzahl

Bereiche Eingangs- und Ausgangsverwiegung bzw. Verwiegung ohne Zuordnung
gesamt

Anzahl der Sätze des jeweiligen Bereichs

doppelte

Anzahl der Sätze, die bei einer Prüfung innerhalb der Zwischentabellen bzw. gegen den Echtbestand als doppelt erkannt wurden

fehlerhafte

Anzahl der Sätze, deren Pflichtfelder nicht gefüllt sind

1.1.10 Ohne Wiegeschein

Anzahl der Sätze, zu denen kein Wiegescheindetail vorliegt

1.1.11 Vollständig

Anzahl der Sätze, bei denen alle Pflichtfelder gefüllt sind, min. ein Wiegeschein

gefunden wurde und die nicht innerhalb der Zwischentabellen und des Echtdatenbestandes doppelt sind.

4.3. Dialog: Waage Schnittstelle Übernahme


In dem Dialog des Menüpunktes **Transfer → Waage Schnittstelle → Daten übernehmen** wird die Übernahme der Daten aus den Zwischentabellen in den Echtdatenbestand ausgelöst.

Abbildung 4-3: Dialog Waage Schnittstelle Übernahme

Der Button „Anzahl ermitteln“ prüft die Anzahl der zu übernehmenden, vollständigen Sätze, bei denen die technischen Voraussetzungen für eine Übernahme gegeben sind und nicht als doppelt erkannt wurden. Der Button „Auswahl übernehmen“ löst den Übernahmevorgang in die Echttabellen aus. Dabei werden die Daten gegen den Echtdatenbestand auf fachlicher Ebene validiert.

Übernommene Daten können mit den Dialogen **Funktionen → Verwiegung → Eingangsverwiegung** bzw. **Funktionen → Verwiegung → Ausgangsverwiegung** nachbearbeitet werden und sind danach buchbar.

Datensätze, die nicht übernommen werden konnten, verbleiben in den Zwischentabellen.

 Zur Übernahme stehen nur vollständige, nicht als doppelt erkannte Verwiegungen zur Verfügung, d.h. Verwiegungsmaster inklusive Wiegescheindetails. Die Informationen zu Kostenstellenaufteilungsdetails sind nicht Voraussetzung für eine Übernahme.

5 Beispiel für die Umsetzung der alten Meldeschnittstelle BA31/32

Die neue Meldeschnittstelle beinhaltet sowohl formale Änderungen gegenüber den alten BA31/BA32 Meldeschnittstellen als auch die sich aus den neuen Verträgen ergebenden fachlichen Änderungen. Diese Änderungen sind im w^me.fact abgebildet. Entsprechend ist die Waageschnittstelle für den Import anders gestaltet als die der alten WME - Schnittstelle.

1.1.12 Fachliche Änderungen:

In der alten WME - Schnittstelle war die Zuordnung der Mengen zu Garantiegebern sowie zu Vertragsgebieten ausreichend. Aufgrund der neuen vertraglichen Regelungen ergibt sich nun die Anforderung, andere bzw. weitere Informationen anzugeben.

Insbesondere sind dies:

- statt der bisherigen Vertragsgebietsinformationen sind den Mengenbewegungen die neuen Leistungsvertragsnummern zuzuordnen, optional können für die Sammelmengen die Kommunen und die dort verwendeten Sammelsysteme angegeben werden
- die neu hinzugekommenen Systembetreiber sind zum Leistungsvertrag zu berücksichtigen (z.B. Landbell)

Diese Informationen werden im Sprachgebrauch der neuen Schnittstelle als Kostenstelleninformationen bezeichnet und in jeder Verwiegung (sprich dem Master der Verwiegung, nicht dem Wiegeschein) mit der Angabe des Anteils in % an der Verwiegung angegeben und übermittelt.

1.1.13 Formale Änderungen:

Die Bereitstellung in der seit 1995 gültigen WME - Schnittstellenbeschreibung wie auch die Datenhaltung zur Verwiegung erfolgte in einer flachen Struktur, d.h. jeder Wiegeschein wurde als ein Satz gemeldet und enthielt alle Zusatzinformationen wie Vertragsgebiet, Garantiegeber, etc.. Lagen mehrere Zuordnungen vor (z.B. wenn der angegebene Wiegeschein auf mehrere Vertragsgebiete aufgeteilt war), wurde der Wiegeschein mehrfach gemeldet. Dies entsprach ebenfalls der verwendeten Übertragungsstruktur eindimensionaler Tabellen.

Demgegenüber steht die neue Struktur (siehe hierzu die Dialoge zur Eingangs- bzw. Ausgangsverwiegung). Es wird zunächst ein Master gebildet, der alle einmal vorkommenden Informationen enthält. Der Master enthält alle wesentlichen Informationen eines Transportes. Einem Master können mehrere Wiegescheine sowie mehrere Kostenstelleninformationen zugeordnet werden.

Diese Struktur wurde in der neuen Waageschnittstelle entsprechend umgesetzt. Die in w^me.fact zu importierenden Daten (evtl. im Format der alten BA31) müssen zunächst im Format des neuen w^me.fact aufbereitet werden. Hierzu soll das nachfolgende Beispiel Hilfestellung geben. Dazu werden die einzelnen w^me.fact - Zielvariablen entsprechend ihres Auftretens in der Schnittstelle nachfolgend einzeln erklärt sowie deren Umsetzung abgeleitet.

Zu beachten ist insbesondere, dass in der BA31/BA32 Schnittstelle die Verwiegungen der unterschiedlichsten Anlagen gemeinsam übertragen werden konnten. w^me.fact hingegen basiert auf der Verwaltung der Daten einer Anlage, d.h. über die Waageschnittstelle können jeweils nur die Bewegungsdaten der importierenden Anlage verarbeitet werden.

Der Auslieferungsversion liegt die hier besprochene Schnittstellendefinition bei (siehe Verzeichnis <wme.fact>Schnittstelle\WsBA31Beispiel). In diesem Beispiel sind in der BA31-file-definition.xml die Felddefinitionen für eine Datendatei entsprechend der alten Schnittstellenbeschreibung hinterlegt. In der BA31-interface.xml sind, soweit als möglich und sinnvoll, diese Quellvariablen direkt den Zielvariablen zugeordnet. Da eine direkte Zuordnung nicht in allen Fällen möglich ist, werden in diesen Fällen die Importdaten über Multi-Mappings zunächst umgeschlüsselt.

5.1. Bereich Master

5.1.1. Auftragsnummern und freie Auftragsnummern

^{wm}e.fact unterscheidet Aufträge aus dem Bereich des DSD und freie Auftragsnummern. Während die Auftragsnummer DSD auf die bereits in der Tabelle TTR_AUFTRAEGE importierten Datensätze referenziert, werden freie Auftragsnummer in der Variable „AUFTRAGSNUMMER“ gespeichert. Auftragsnummern können demnach nur zusammen mit der Auftragsart („DSD“ oder „intern“) eindeutig zugeordnet werden.

Das Beispiel enthält keine zu importierenden DSD - Aufträge, daher wird für die Auftragsnummer beispielhaft die Nummer des Wiegescheins herangezogen. Alternativ könnte auch eine beliebige andere Zahl oder Buchstabenkombination verwendet werden.

Syntax BA31-interface.xml:

```
<variable ref = „AUFTRAGSNUMMER“ field = „WIEGE_NR“/>
```

Für den Import von DSD - Aufträgen müsste die oben angegebene Syntax wie folgt ersetzt werden:

```
<variable ref="AUFTRAG">
```

```
<get-id-from container="AUFTRAG">
```

```
  <variable ref="v_Auftragsnummer" field="!!SPALTENNAME!!"/>
```

```
  <variable ref="n_AuftragsartID_Auftra">
```

```
    <get-id-from container="AUFTRAGSART">
```

```
<variable ref="v_ArtBezeichnung" field="!!SPALTENNAME!!"/>
```

```
  </get-id-from>
```

```
</variable>
```

```
<variable ref="n_GarantiegeberID_Persk">
```

```
  <get-id-from container="PERSONENKONTO">
```

```
    <variable ref="v_KontoNR" field="!!SPALTENNAME!!"/>
```

```
    <variable ref="n_KontoklasseID_Kokl">
```

```
<get-id-from container="KONTOKLASSE">
```

```
<variable ref="v_Bezeichnung" field="!!SPALTENNAME!!"/>
```

```
</get-id-from>
```

```
</variable>
```

```
  </get-id-from>
```

```
</variable>
```

```
  </get-id-from>
```

```
</variable>
```

Auftragsart muss DSD sein (Kennnummer = 1, Bezeichnung = DSD). Die Kontonummer des Garantiegebers ist in diesem Fall 3, die Klasse ist „Garantiegeber“, siehe [Anhang 11.4.5.](#) und [Anhang 11.4.16.](#)

5.1.2. Buchungsarten

Syntax BA31-interface.xml:

```
<variable ref="BUCHUNGSART">
```

```
<get-id-from container="BUCHUNGSARTEN">
```

```
<variable ref="n_Kennnummer" multi-mapping="buchungsart-alias/BUCHUNGSART_KENNR"/>
```

```
</variable>
```

Anzugeben ist die Kennnummer der Buchungsart. Die Buchungsart klassifiziert im Sprachgebrauch der Waageschnittstelle die Sender-/ Empfängertypisierung. Unterschieden werden dabei grundsätzlich Wareneingänge und Warenausgänge. Diese waren in der alten Schnittstelle gekennzeichnet durch das Feld „Belegart“ („BA31“ = „Wareneingang“, „BA32“ = „Warenausgang“).

Allerdings ist diese einfache Unterscheidung nicht mehr ausreichend. So werden z.B. die Buchungsarten zur Eingangsverwiegung (EV) weiter differenziert nach „EV aus Haushaltssammlung“, „EV aus Gewerbesammlung“, „EV von Umschlaganlage“, etc. (siehe. auch [Anhang 11.4.4.](#)) Anhand der einzelnen Buchungsarten erfolgen verschiedene Validierungen innerhalb des ^{wm}e.fact.

Die einzelnen Buchungsarten innerhalb der EV/AV können über den Dialog **Funktionen → Verwiegung → Eingangsverwiegung** bzw. **Funktionen → Verwiegung → Ausgangsverwiegung** in ^{wm}e.fact nachvollzogen werden. Alternativ können die Buchungsarten über die Tabelle TYF_BUCHUNGSARTEN in der Datenbank mebus.gdb ausgelesen werden.

Für die Zuweisung eines Zielwertes, also der Kennnummer der Buchungsart, bieten sich damit verschiedene Möglichkeiten:

- Können alle Daten einer Importroutine eindeutig einer Buchungsart zugewiesen werden, so kann dieser Wert als default - Wert hinterlegt werden.
- Unter Berücksichtigung des Feldinhaltes der Variable „Herkunftsanlage“ der BA31 besteht die Möglichkeit der Ableitung der Buchungsart, d.h. über eine Umschlüsselung kann die Buchungsortung automatisiert ermittelt werden.

Keine Herkunftsanlage:

Ist keine Herkunftsanlage genannt, so kann es sich bei dem Wareneingang nur um eine Eingangsverwiegung aus Sammlung handeln. Mögliche Buchungsarten sind „EV aus Haushaltssammlung“ oder „EV aus Gewerbesammlung“.

Eine Herkunftsanlage:

Ist eine Herkunftsanlage genannt, so bestimmt die Kontoklasse der Herkunftsanlage die Buchungsart (wenn z.B. die Herkunftsanlage eine Sortieranlage ist (Kontoklasse = Sortieranlage), wird die Buchungsart „EV von Sortieranlage“ verwendet). Dieser Ansatz ist im vorliegenden Beispiel gewählt worden und findet sich wieder in Umschlüsselungstabelle „buchungsart-mm.xml“. Für die Verwendung ist die Multi-Mapping-Tabelle mit Werten zu füllen. Analog kann die Ableitung beim Warenausgang / BA32 erfolgen.

- Möglicherweise kann die Buchungsart aber auch aus anderen Informationen abgeleitet werden, wenn z.B. die Fraktion der BA31 an der vorhandenen Anlage bereits eine eindeutige Ableitung erlaubt. Hierzu muss eine entsprechende Multi-Mapping-Tabelle angelegt werden.

1.1.14 Abhängigkeiten

Die Variable „Eingangsverwiegung“ als Boolean-Feld muss entsprechend der Buchungsart richtig gesetzt werden (T für Buchungsarten EV, F für Buchungsarten AV); (siehe. auch [Anhang 11.4.4.](#)).

Aus der Buchungsart und der Kontoklasse der importierenden Anlage ergeben sich Abhängigkeiten für die Kennzeichnung von Retourenkennzeichen und Retourengrund (siehe. auch [Anhang 11.4.18.](#)).

5.1.3. Personenkonten und Vertragspartner

Syntax BA31-interface.xml:

```
<variable ref="VERTRAGSPARTNER">

<get-id-from container="PERSONENKONTO">
<variable ref="v_KontoNR" multi-mapping="buchungsart-
alias/VERTRAGSPARTNER_KTONR"/>
<variable ref="n_KontoklasseID_Kokl">

<get-id-from container="KONTOKLASSE">
<variable ref="v_Bezeichnung" multi-mapping="buchungsart-
alias/VERTRAGSPARTNER_KTOKLASSE"/>
</get-id-from>
</variable>
</get-id-from>
</variable>
```

In der Syntax der alten WME - Schnittstelle wird unterschieden nach

- **Belieferte Anlage (ANL_NR_BEL)**
Anlage, an dem der Vorgang stattfindet;
 - ⚠ entspricht in w^me.fact der Anlage (siehe auch nächstes Kapitel)
- **Herkunftsanlage (ANL_NR_HER)**
Senderanlage, von der die Mengen angeliefert wurden. Diese Information wurde nur gesetzt, wenn die unmittelbare Herkunft der Mengen nicht direkt aus der Sammlung stammten und somit einen Anlagen-Anlagen-Transport vor der Schnittstelle darstellte; ansonsten blieb dieses Feld leer
 - ⚠ bei einer Eingangsverwiegung von Mengen in w^me.fact ist der Vertragspartner immer ein Personenkonto; je nach Buchungsart sind nur Personenkonten bestimmter Kontoklassen zulässig
- **Empfangende Anlage (EMPFAENGER)**
Beim einem Output mit der Zuständigkeit „Garantiegeber“ wurde die Anlagennummer des Empfängers angegeben
 - ⚠ bei einer Ausgangsverwiegung von Mengen ist jetzt immer ein Empfänger anzugeben; je nach Buchungsart sind nur Personenkonten bestimmter Kontoklassen zulässig

Im Sprachgebrauch der neuen w^me.fact - Schnittstelle werden alle Vertragspartner und Anlagen zunächst als Personenkonten bezeichnet. Zusammen mit der Kontoklasse (z.B. Anlieferer oder Sortieranlage) werden diese eindeutig. Im Kontext von Buchungsart (EV von Sortieranlage) und Verwiegung (Eingangsverwiegung oder Ausgangsverwiegung) und unter Berücksichtigung der Kontoklasse der importierenden Anlage (= Kontoklasse) wird nun nicht mehr auf die Unterscheidung nach belieferte Anlage / Herkunftsanlage etc. abgehoben, sondern nur noch der jeweils in der Verwiegung beteiligte Vertragspartner genannt.

1.1.15 Beispiel:

Bei einer Eingangsverwiegung von Sortieranlage (EV von Sortieranlage) an einer Veredelungsanlage (= Kontoklasse der importierenden Anlage) muss der Vertragspartner eine Sortieranlage sein.

5.1.4. Belieferte Anlagen

Bisher konnten in WME die Daten einer Vielzahl von Anlagen gesamthaft verwaltet werden. Im neuen Konzept des w^me.fact wird jede Anlage getrennt betrachtet. Der Import von Daten aus mehreren Anlagen in einem einzigen Import ist nicht möglich. Filterkriterien stehen nicht zur Verfügung.

Die vorliegenden Importdaten im Format BA31 (data\BEISPIEL_BA31.321) genügen nicht den Anforderungen von w^me.fact, teilweise sind Felder neu hinzugekommen, teilweise haben sich Inhalte geändert. Deshalb muss in weiten Teilen mit Multi-Mappings eine Umschlüsselung der alten Daten erfolgen. Für die Beispieldaten wurde als zentrales Multi-Mapping folgender Ansatz gewählt:

```

27      <record>
28          <source name="BA_NR_ALT">31</source>
29          <source name="ANL_NR_BEL_ALT">11111</source>
30          <source name="ANL_NR_HER_ALT">0</source>
31          <source name="DUPLIZIER_ALT">F</source>
32          <source name="STO_GRD_ALT">00</source>
33
34          <target name="BUCHUNGSART_KENNR">100</target>
35          <target name="WARENEINGANG">T</target>
36          <target name="SENDERWIEGESCHEIN">F</target>
37          <target name="VERTRAGSPARTNER_KTONR">48114</target>
38          <target name="VERTRAGSPARTNER_KTOKLASSE">Anlieferer</target>
39          <target name="GG_KTONR">48117</target>
40          <target name="GG_KTOKLASSE">Garantiegeber</target>
41      </record>

```

Abbildung 5-1: Multi - Mapping buchungsart-mm.xml (Ausschnitt)

Dieses Mapping liest sich wie folgt:


Wenn

- die BA_NR = 31 ist (= Input), UND
- die belieferte Anlage = 11111, UND
- die Mengen nicht aus einer anderen Anlage stammen, UND
- das Flag Duplizier = false ist, UND
- der Datensatz kein Storno ist

Dann

- setze die Buchungsart = 100 (Wareneingang aus Haushaltssammlung)
- setze Wareneingang = true
- setze Senderwiegeschein = false (ist also ein Empfängerwiegeschein)
- setze die Kontonummer des Vertragspartners = 48114 (= Sonstige Anlieferer)
- setze die Kontoklasse des Vertragspartners = Anlieferer
- setze die Kontonummer des Garantiegebers = 48117 (= kein Garantiegeber)
- setze die Kontoklasse des Garantiegebers = Garantiegeber

Damit sind bereits wesentliche Felder des Imports gesetzt.

 Dieses Multi-Mapping ist natürlich stark auf die mitgelieferten Beispieldaten ausgelegt. Sollten andere Daten damit importiert werden, muss ggf. das Multi-Mapping angepasst werden.

5.1.5. Fraktion zu Artikel, Lagerorte

Syntax BA31-interface.xml:

```

<variable ref="ARTIKEL">

<get-id-from container="ARTIKEL">
    <variable ref="v_Artikelnummer" field="FRAKT_NR"/>
</get-id-from>
</variable>
<variable ref="ARTIKELLAGERORT">
    <get-id-from container="ARTIKELLAGERORT">
<variable ref="n_ArtikelID_Artk">
    <get-id-from container="ARTIKEL">
<variable ref="v_Artikelnummer" field="FRAKT_NR"/>
    </get-id-from>
    </variable>
    <variable ref="n_LagerortID_Lago">
    <get-id-from container="LAGERORT">
<variable ref="v_Kurzname" multi-mapping="lagerorte-alias/LAGERORT_KURZNAME"/>
    </get-id-from>
</variable>
    </get-id-from>
</variable>

```

Die bisherigen WME Fraktionen wurden in Hauptartikelgruppen umbenannt und um Artikelgruppen (Feinfraktionen) erweitert. Die Artikel als führende Größe sind Artikelgruppen zugeordnet. Neu hinzugekommen sind die Lagerorte. Mengen eines Artikels können nur auf zugeordnete Lagerorte verbucht werden.

Zur näheren Erläuterung wird der Artikel zunächst mit seinen Entitäten und Referenzen in der Datenbankstruktur kurz dargestellt.

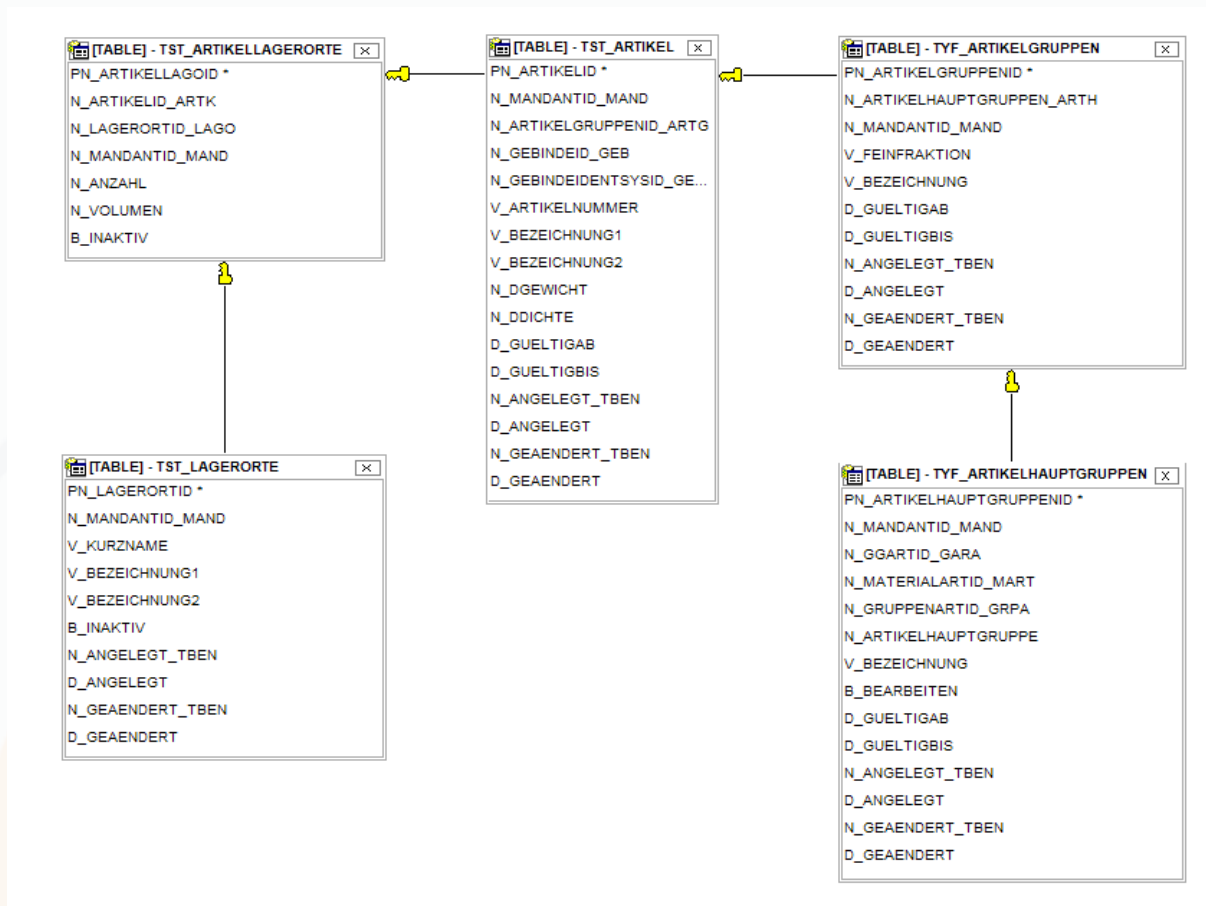



Abbildung 5-2: Darstellung Abhängigkeiten Artikel

Für die Waageschnittstelle sind nur der Artikel und der Lagerort von Interesse. Die Zuordnung zu den DSD-Feinfraktionen ergibt sich durch die in den Stammdaten erfolgten Zuordnungen.

Zudem werden Artikel als Volumen- oder Stückartikel geführt (N_GEBINDEID_GEB). Zu jeder Mengenbewegung müssen die Mengen in Gebinde oder Volumen mit angegeben werden. Dies führt in der Umkehrung dazu, dass bei einem Importsatz mit einer Mengenangabe „Volumen“ ein Volumenartikel gewählt werden muss, ansonsten ist ein Stückartikel zu wählen.

 Werden beim Import in der Schnittstellendefinition weder Volumen noch Anzahl angegeben, so wird der dem Artikel entsprechende Wert (entweder Volumen oder Anzahl) bei der Übernahme in den Echtdatenbestand automatisch berechnet. Die Berechnung erfolgt über die in ^{wm}e.fact zu dem Artikel hinterlegten, durchschnittlichen Gewichte.


Im Beispiel wurde der Ansatz gewählt, für jede Fraktion die importiert werden soll, einen Artikel mit der Artikelnummer gleich der Fraktionsnummer anzulegen. Da das Konzept der Lagerorte neu ist, wird der Lagerort zu einem Artikel über Multi - Mapping bestimmt, im vorliegenden Beispiel über die lagerorte-mm.xml. Darin erfolgt eine einfache Zuordnung einer Fraktionsnummer zu einem Lagerort, dies kann durchaus immer derselbe Lagerort sein.

5.1.6. Eingang/Ausgang

Syntax BA31-interface.xml:

```
<variable ref="EINGANG" multi-mapping="buchungsart-alias/WARENEINGANG"/>
```

Analog zu dem Feld BA_NR der alten WME - Schnittstelle, gibt es in w^me.fact eine Variable Eingang. Diese kann belegt werden mit den Werten T (= true) oder F (= false). T bedeutet Wareneingang und entspricht dem Wert 31 des Feldes „BA_NR“. F bedeutet Warenausgang und entspricht dem Wert 32 des Feldes „BA_NR“. Somit ist die Bedeutung identisch, allerdings nicht der Inhalt. Deshalb erfolgt die Belegung der Variablen EINGANG über ein Multi-Mapping.

 Eingangsverwiegungen wurden in der Vergangenheit im DSD Sprachgebrauch als INPUT und Ausgangsverwiegungen als OUTPUT bezeichnet.

5.1.7. Retoure

Syntax BA31-interface.xml:

```
<variable ref="RETOURE" >>
  <get-id-from container="RETOUREN">
<variable ref="n_Kennnummer" multi-mapping="retoure-alias/RETOURE_KENNR"/>
  </get-id-from>
</variable>
```

Die Retourinformation dient primär zur genauen Ermittlung der vertraglich erbrachten Liefermengen; d.h. es werden nur die Retouren abgegriffen, die an der Schnittstelle zwischen Sortierung und Verwertung entstanden sind und dem Liefermengenkonto des Vertragspartners abzuziehen sind. Retouren innerhalb einer Anlagengruppe werden nicht gesondert als Retoure gekennzeichnet.

Für Retoure.n_Kennnummer kann entweder 1 (= keine Retoure) oder 2 (= Nicht spezifikationsgerecht) angegeben werden (siehe [Anhang 11.4.2.](#)). Für den Zusammenhang zwischen Buchungsart, Retourangabe und Kontogruppe siehe [Anhang 11.4.18.](#)

5.1.8. Garantiegeber

Syntax BA31-interface.xml:

```
<variable ref="GARANTIEGEBER">
  <get-id-from container="PERSONENKONTO">
    <variable ref="v_KontoNR" multi-mapping="buchungsart-alias/GG_KTONR"/>
    <variable ref="n_KontoklasseID_Kokl">
      <get-id-from container="KONTOKLASSE">
<variable ref="v_Bezeichnung" multi-mapping="buchungsart-alias/GG_KTOKLASSE"/>
      </get-id-from>
</variable>
    </get-id-from>
</variable>
```

Angabe eines Garantiegebers, der den weiteren Verwertungsnachweis führt. Die anzugebenden Garantiegeber werden über die Buchungsart, die Artikelhauptfraktion und die Anlagengruppe festgelegt. Mögliche Bezeichnungen für Kontoklassen sind unter [Anhang 11.4.5.](#) aufgeführt; eine Liste aller Garantiegeber sind dem [Anhang 11.4.16.](#) zu entnehmen.

5.2. Bereich Wiegeschein

5.2.1. Sender-/Empfängerwiegeschein

Syntax BA31-interface.xml:

```
<variable ref="SENDERWS" multi-mapping="buchungsart-alias/SENDERWIEGESCHEIN"/>
```

w^me.fact unterscheidet bei Wiegescheinen, ob es sich um Sender- oder Empfängerwiegescheine handelt.

Definition:

- Senderwiegeschein: Beleg wurde an der Senderanlage erstellt
- Empfängerwiegeschein: Beleg wurde an der Empfängeranlage erstellt

Sind sowohl Empfängerwiegeschein(e) als auch Senderwiegeschein(e) erfasst, so wird bei einer Eingangsverwiegung der Empfängerwiegeschein und bei einer Ausgangsverwiegung der Senderwiegeschein als abrechnungsrelevanter Beleg herangezogen.

Derzeit können bei Einzelverwiegungen eines Transportes max. 3 Wiegescheine an der Sender- und/oder Empfängeranlage erstellt werden wie z.B.:

- Transport von Material mit 1 Containerfahrzeug mit 2 Hängerfahrzeugen

In der BA31 wurde jeder Wiegeschein in einem eigenen Satz dargestellt. Sollen diese drei Sätze (aus o.g. Beispiel) einem Verwiegungsmaster zugeordnet werden, so kann dies über Verwendung der Group - Funktionalität bei den Variablen der TIM_VERWIEGUNG erfolgen (siehe [Kapitel 8](#)).

Der Import von sowohl Sender- als auch Empfängerwiegescheinen ist z.B. dann sinnvoll, wenn die gesamte Transportinformation an weitere Nachweisstellen (z.B.: Garantiegeber) zur Verfügung gestellt werden soll. Die Verfahrensweise ist unter den Beteiligten abzustimmen.

Sofern nur ein Wiegeschein zur Dokumentation des Transportes erstellt wurde, wird dieser sowohl für den Sender als auch für den Empfänger als abrechnungsrelevanter Beleg zu Grunde gelegt.

Beispiel: Bei der Senderanlage wurde ein Wiegebeleg erstellt, bei der Empfängeranlage ist die Waageeinrichtung ausgefallen.

Mögliche Werte für die Variable SENDERWS sind T (= true =Senderwiegeschein) oder F (= false =Empfängerwiegeschein).

5.2.2. Verwiegungs-Status

Syntax BA31-interface.xml:

```
<variable ref="VWSTATUS">
```

```
<get-id-from container="VWSTATUS">
```

```
<variable ref="n_Kennnummer" default="1"/>
```

```
</get-id-from>
```

```
</variable>
```

Dokumentiert die Art der Erstellung des Wiegebeleges, z. B. Elektronischer Beleg oder Elektronischer Beleg mit nachträglicher Gewichtsänderung. Eine Übersicht möglicher Kennnummern sind unter [Anhang 11.4.3.](#) zusammengestellt.

5.2.3. Nettokorrektur

Syntax BA31-interface.xml:

```
<variable ref="NETTOKORREKTUR" default="0"/>
```

Die Nettokorrektur führt zu einer Reduzierung der eingewogenen Nettotonnage. Der Korrekturwert ist als positiver Zahlenwert anzugeben (≥ 0). Nettokorrekturen sind z.B. dann vorzunehmen, wenn anteilige Mengen, die nicht dualen Systemen zugeordnet werden dürfen, gemeinsam mit Mengen aus dualen Systemen transportiert und verworfen wurden, wie z.B. gewerbliche Mengen oder Pfandmengen. Ebenfalls ist eine Nettokorrektur dann vorzunehmen, wenn Transporthilfsmittel eingesetzt und mit verworfen wurden (z.B. Paletten etc.). Im vorliegenden Fall wird die NETTOKORREKTUR immer mit 0 belegt, da die Beispieldaten keine Fremdmengen enthalten.

5.2.4. KFZ zu KFZ

KFZ - Kennzeichen des alten WME müssen aufgeteilt nach Land – Kreis – KFZ - Kennzeichen bereitgestellt werden. Ein Länderkennzeichen muss angegeben werden. Da in der alten WME - Schnittstelle diese Aufteilung nicht gefordert war, wird im vorliegenden Beispiel das KFZ - Kennzeichen über Reguläre Ausdrücke aufgeteilt.

Beispiel zu 5.9.:

```
189      <!-- gibt es nicht in BA31, wird aber benötigt für Übernahme -->
190      <variable ref="NETTOKORREKTUR" default="0"/>
191
192      <!-- Status Standardmäßig auf "Verwiegung" -->
193      <variable ref="VWSTATUS">
194          <get-id-from container="VWSTATUS">
195              <variable ref="n_Kennnummer" default="1"/>
196          </get-id-from>
197      </variable>
198
199      <!-- gibt es nicht in BA31, wird aber benötigt für Übernahme -->
200      <variable ref="LAND">
201          <get-id-from container="LAND">
202              <variable ref="v_Laenderkennzeichen" default="D"/>
203          </get-id-from>
204      </variable>
205
206      <variable ref="KFZ">
207          <get-id-from container="KREIS">
208              <variable ref="v_Kuerzel" field="KFZ">
209                  <filter>
210                      <!-- alles raus, was nicht in ein Kenzeichen gehört -->
211                      <regexp pattern="[_-\s]" replacement=" '/>
212                      <regexp pattern="[^\w ]" replacement=""/>
213                      <!-- alles bis zum ersten Blank ist Kreis -->
214                      <regexp pattern="(.*?) [ \d].*" replacement="\U$1' />
215                  </filter>
216              </variable>
217          </get-id-from>
218      </variable>
219
220      <variable ref="KENNZEICHENFREI" field="KFZ">
221          <filter>
222              <!-- alles raus, was nicht in ein Kenzeichen gehört -->
223              <regexp pattern="[_-\s]" replacement=" '/>
224              <regexp pattern="[^\w ]" replacement=""/>
225              <!-- alles ab dem ersten Blank / der ersten Ziffer extrahieren -->
226              <regexp pattern=".*?([ \d].*)" replacement="$1' />
227              <!-- Buchstaben und Ziffern ohne sonstige Zeichendazwischen -->
228              <regexp pattern="([^\d ]*) *(\d*)" replacement="$1$2' />
229              <!-- links und rechts trimmen und Uppercase-->
230              <regexp pattern="^\s*(.*?)\s*$" replacement="\U$1' />
231          </filter>
232      </variable>
```

Abbildung 5-3: BA31-interface.xml (Ausschnitt)

Das Länderkennzeichen wird im Beispiel standardmäßig mit „D“ für Deutschland belegt, da die Beispieldaten keine ausländischen Anlieferungen enthalten.

Das Beispiel setzt voraus, dass die KFZ - Kennzeichen in den Importdaten gemäß der alten WME – Schnittstellendefinition korrekt angegeben wurden, d.h., dass zwischen dem Kreis und dem Rest des Kennzeichens ein Trennungszeichen verwendet wurde (in der Regel ein Bindestrich). Sollte dies nicht der Fall sein, so besteht keine Möglichkeit dieses Kennzeichen korrekt aufzulösen.

Z.B. KFZ = BNM123 könnte sowohl B-NM123 (Berlin), als auch BN-M123 (Bonn) sein.

5.3. Bereich Kostenstellenanteile

5.3.1. Kostenstellenfall

Syntax BA31-interface.xml:

```
<variable ref="KOSTENSTELLENFALL">
  <get-id-from container="KOSTENSTELLENFALL">
<variable ref="n_Kennnummer" multi-mapping="kostenstellenfall-alias/KSTFALL_KENNR"/>
  </get-id-from>
</variable>
```

Wie bereits Eingangs erwähnt, sind mit der Einführung von w^me.fact Angaben zu Systembetreibern und optional Kommunen nötig/möglich. Diese Angaben können über die Waageschnittstelle im Bereich Kostenstellenanteile importiert werden. Die Variable KOSTENSTELLENFALL dient zur Angabe, um welchen Kostenstellenfall es sich handelt. Mögliche Werte für n_Kennnummer sind 1 (= LV / Kommune), 2 (= LV / Systembetreiber) oder 6 (= LV / Rücknahmesystem). Der mögliche Kostenstellenfall wird durch die Buchungsart und die Anlagengruppe bestimmt. Die komplette Übersicht ist [Anhang 11.4.21.](#) zu entnehmen.

Zu beachten ist, dass die Verwiegungen zunächst auch ohne Kostenstelleninformationen importiert und diese nach der Übernahme in den Echtbestand mittels der Funktionalität **Funktionen → Verwiegung → Kostenstellenaufteilung erfassen** für eine Vielzahl von Verwiegungen in einem Arbeitsschritt angegeben werden können.

5.3.2. Kostenstelle1

Syntax BA31-interface.xml:

```
<variable ref="KOSTENSTELLE1">

<get-id-from container="KOSTENSTELLE">
  <variable ref="v_Kuerzel" multi-mapping="kostenstellenfall-alias/KST1_KUERZEL"/>
  <variable ref="d_GueltigAb" multi-mapping="kostenstellenfall-alias/KST1_GUELTIGAB"
    format="dd.MM.yy"/>
  <variable ref="n_KostenstartID_Kosta">

<get-id-from container="KOSTENSTELLENART">
<variable ref="v_Bezeichnung" multi-mapping="kostenstellenfall-alias/KST1_ART"/>
  </get-id-from>
</variable>
  </get-id-from>
</variable>
```

Die Kostenstelle1 beinhaltet unabhängig vom Kostenstellenfall immer die Angabe der Leistungsvertragsnummer. Drei Variablen müssen dazu gefüllt werden:

- v_Kuerzel: unabhängig vom Kostenstellenfall, siehe [Anhang 11.4.22](#)
- d_GueltigAb: siehe [Anhang 11.4.22](#)
- n_KostenstartID_Kosta.v_Bezeichnung: siehe [Anhang 11.4.22](#)

5.3.3. Kostenstelle2

Syntax BA31-interface.xml:

```
<variable ref="KOSTENSTELLE2">

  <get-id-from container="KOSTENSTELLE">
    <variable ref="v_Kuerzel" multi-mapping="kostenstellenfall-alias/KST2_KUERZEL"/>
    <variable ref="d_GueltigAb" multi-mapping="kostenstellenfall-alias/KST2_GUELTIGAB"
      format="dd.MM.yy"/>
    <variable ref="n_KostenstartID_Kosta">

  <get-id-from container="KOSTENSTELLENART">
    <variable ref="v_Bezeichnung" multi-mapping="kostenstellenfall-alias/KST2_ART"/>
    </get-id-from>
  </variable>
  </get-id-from>
</variable>
```

In Abhängigkeit des Kostenstellenfalls muss hier ein Systembetreiber bzw. eine Kommune oder ein Rücknahmesystem angegeben werden. Drei Variablen müssen dazu gefüllt werden:

- v_Kuerzel: abhängig vom Kostenstellenfall, siehe [Anhang 11.4.23](#)
- d_GueltigAb: siehe [Anhang 11.4.23](#)
- n_KostenstartID_Kosta.v_Bezeichnung: siehe [Anhang 11.4.23](#)

5.3.4. Kostenstelle3

Syntax BA31-interface.xml:

```
<variable ref="KOSTENSTELLE3">
  <get-id-from container="KOSTENSTELLE">
    <variable ref="v_Kuerzel" multi-mapping="kostenstellenfall-alias/KST3_KUERZEL"/>
    <variable ref="d_GueltigAb" multi-mapping="kostenstellenfall-alias/KST3_GUELTIGAB"
      format="dd.MM.yy"/>
  <variable ref="n_KostenstartID_Kosta">
    <get-id-from container="KOSTENSTELLENART">
  <variable ref="v_Bezeichnung" multi-mapping="kostenstellenfall-alias/KST3_ART"/>
  </get-id-from>
  </variable>
  </get-id-from>
</variable>
```

In Abhängigkeit des Kostenstellenfalls muss hier „ohne“ bzw. ein Sammelsystem angegeben werden. Drei Variablen müssen dazu gefüllt werden:

- v_Kuerzel: abhängig vom Kostenstellenfall, siehe [Anhang 11.4.24](#)
- d_GueltigAb: siehe [Anhang 11.4.24](#)
- n_KostenstartID_Kosta.v_Bezeichnung: siehe [Anhang 11.4.24](#)

6 Detaillierte Ausführungen

6.1. -import.xml

Die -import.xml ist die „Einstiegsdefinition“ des Imports, und wird über folgende Namens-Konvention im Unterverzeichnis <wme.fact>\Schnittstelle gefunden: <Routinenname>\<Routinenname>-import.xml.

<import-task

```
xmlns="http://www.dsd-ag.de/wme-fact/transfer/task"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/task
file:///c:/programme/wme.fact/Config/Transfer/schema/task.xsd"
name="Name der Routine"
transaction="true">
```

```
<interface href="Pfad und Name der interface.xml">
<file href="Pfad und Name der zu importierenden Datei"/>
<param name="globale Variable" value="globaler Wert"/>
</interface>
```

</import-task>

6.1.1. Wurzel-Element <import-task>

<import-task>		Default	Pflicht	Anzahl
xmlns	Der Default-Namespace dieses Instanzdokumentes. Muss hier immer „http://www.dsd-ag.de/wme-fact/transfer/task“ lauten.		ja	
xmlns:xsi	Der sog. „XML Schema instance namespace“. Wird nur gebraucht, um das nächste Attribut – xsi:schemaLocation – angeben zu können. In diesem Fall muss stets „ http://www.w3.org/2001/XMLSchema-instance “ gesetzt sein.		(Ja),	
xsi:schemaLocation	Mit Hilfe dieses Attributes wird einem Namespace dessen Schemadefinition (.xsd-Datei) zugeordnet. w ^m e.fact selbst ignoriert dieses Attribut, da es die Zuordnung „kennt“, d.h. es weiß, wo es die Schemadefinition zu diesem Namespace zu suchen hat. Jedoch kann dieses Attribut z.B. XML-Editoren als Hinweis dienen, wo die Schemadefinition zu finden ist. Anhand dieser Schemadefinition kann der Editor dann beispielsweise die jeweils erlaubten Elemente oder Attribute vorschlagen sowie die XML Datei auf Gültigkeit prüfen.		Nein	
name	Entspricht dem Wert aus Routine		Ja	
transaction	Vorgabe der Transaktionsklammer für diese Importroutine	true	Nein	
<interface>			Nein	0...n

Hinweis: Die Attribute xmlns, xmlns:xsi und xsi:schemaLocation sind nicht w^me.fact spezifisch, sondern im XML Standard definiert. Für weiterführende Informationen sei auf die relevanten „Recommendations“ des W3C verwiesen:

- <http://www.w3.org/TR/REC-xml>
- <http://www.w3.org/TR/xmlschema-1>
- <http://www.w3.org/TR/REC-xml-names>

Das Attribut *transaction* ist im Kapitel Transaktionsmanagement (siehe [Kapitel 7](#)) beschrieben.

6.1.2. Element <interface>

<interface>		Default	Pflicht	Anzahl
href	Pfad und Name der -interface.xml		ja	
Id	Mögliche ID eines InterfaceCase		nein	
<file>			ja	1
<param>			nein	0...n


6.1.3. Element <file>

<file>		Default	Pflicht	Anzahl
href	Pfad und Name der Import-Datendatei		ja	
mandatory	Importdatei muss existieren	true	nein	

6.1.4. Element <param>

<param>		Default	Pflicht	Anzahl
name	Name des Parameters		ja	
value	Wert des Parameters		ja	

Das Element <param> dient der Definition und Wertzuweisung von globalen Variablen für die gesamte Routine. Diese können in der Schnittstellen-Definition (-interface.xml) verwendet werden. In der Schnittstellen-Definition können diese Parameter mit %%NAME%% referenziert werden. Damit lässt sich die -interface.xml so parametrisieren, dass sie z.B. für mehrere Importdateien ohne Modifikation genutzt werden kann.

 Das Element <interface> kann beliebig oft wiederholt werden. Es ist also möglich, den Import auf mehrere Import-Datendateien aufzuteilen, z.B. Daten der Verwiegungsmaster in Datei A, Daten der Wiegescheine in Datei B. Für jede Datei würde ein eigenes <interface> - Element angegeben, welches über sein Kindelement <file> auf die jeweilige Datei A, bzw. B referenziert. In einem solchen Fall sind natürlich auch 2 entsprechende -interface.xml zu konfigurieren.

Über den Dialog „Import“ kann der Benutzer eine oder mehrere Schnittstellen anlegen. Ausschlaggebend ist, welche Daten importiert werden sollen:

- Enthält die Importdatei eine Vielzahl von Informationen, um sowohl Verwiegungsmaster als auch Wiegescheindetails und Kostenstellenanteile in einem Importschritt füllen zu können, so wird der Bereich „Verwiegungen (komplett)“ in der Schnittstelle ausgewählt. Entsprechend wird nur ein -interface.xml - Template angelegt.
- Sollen in einer Importroutine verschiedene Importdateien für einen Importvorgang angesprochen werden (z.B. liegt eine Importdatei mit Informationen für den Verwiegungsmaster und eine Importdatei für Wiegescheindetails vor), so kann dies entweder über zwei verschiedene Importroutinen oder alternativ über eine Importroutine mit zwei Details definiert und verarbeitet werden. Wird eine Importroutine mit zwei Schnittstellen-Details angelegt, werden entsprechend zwei -interface.xml - Templates erzeugt, auf die in der -import.xml referenziert wird.

6.2. -interface.xml

Im Dialog **Transfer** → **Waage Schnittstelle** → **Daten importieren** wurde für den Bereich „Verwiegungen (komplett)“ ein Beispiel für das Template `<Bezeichnung>-interface.xml`, angelegt. Zu beachten ist, dass die folgende Darstellung hier auf die Nennung der Variablen gekürzt wurde.

```

<interface
  xmlns="http://www.dsd-ag.de/wme-fact/transfer/interface"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/interface
    file:///D:/wme.fact/Config/Transfer/schema/interface.xsd"
  name="Bezeichnung"
  container="Wareneinausgang">
  <file-definition href="Bezeichnung-file-definition.xml"/>
  <import-schema typeFormattingRuleset="wme.fact">
    <import-container ref="TIM_VERWIEGUNG">
      <variable ref="BUCHUNGSART">
      <variable ref="VERTRAGSPARTNER">
      <variable ref="AUFTRAG">
      <variable ref=" »ARTIKEL »">
      <variable ref=" »RETOURE »">
      <variable ref="USERWERT">
      <variable ref="GARANTIEGEBER">
      <variable ref="AUFTRAGSNUMMER" field="!!SPALTENNAME!!"/>
      <variable ref="BELEGDATUM" field="!!SPALTENNAME!!" format="dd.MM.yy"/>
      <variable ref="EINGANG" field="!!SPALTENNAME!!"/>
    <import-container ref="TIM_WIEGESCHEIN">
      <variable ref="WIEGESCHEIN_NR" field="!!SPALTENNAME!!"/>
      <variable ref="VWSTATUS">
      <variable ref="BEFOERDERUNGSART">
      <variable ref="KFZ">
      <variable ref="KENNZEICHENFREI" field="!!SPALTENNAME!!"/>
      <variable ref="DATUM" field="!!SPALTENNAME!!" format="dd.MM.yy"/>
      <variable ref="BRUTTO" field="!!SPALTENNAME!!"/>
      <variable ref="BRUTTOZEIT" field="!!SPALTENNAME!!"/>
      <variable ref="TARA" field="!!SPALTENNAME!!"/>
      <variable ref="TARAZEIT" field="!!SPALTENNAME!!"/>
      <variable ref="NETTOKORREKTUR" field="!!SPALTENNAME!!"/>
      <variable ref="SENDERWS" field="!!SPALTENNAME!!"/>
      <variable ref="ANZAHL" field="!!SPALTENNAME!!"/>
      <variable ref="VOLUMEN" field="!!SPALTENNAME!!"/>
    </import-container>
    <import-container ref=" »KOSTENSTELLENANTEIL »">
      <variable ref="ANTEIL" field="!!SPALTENNAME!!"/>
      <variable ref="KOSTENSTELLENFALL">
      <variable ref="KOSTENSTELLE1">
      <variable ref="KOSTENSTELLE2">
      <variable ref="KOSTENSTELLE3">
    </import-container>
  </import-container>
</import-schema>
<master container/>
<multi-mapping />
</interface>

```

Wurde im Dialog **Transfer → Waage Schnittstelle → Daten importieren** im Feld „Bereich“ ein anderer als der Bereich „Verwiegungen (komplett)“ gewählt, so enthält die -interface.xml nur den Import-Container für den jeweiligen Bereich:

- „Verwiegungen (nur Master)“ = Import-Container „TIM_VERWIEGUNG“
 - ⚠ In der -interface.xml ist nur der Import-Container TIM_VERWIEGUNG angelegt; entsprechend können nur die Master der Verwiegungen importiert werden.
- „Verwiegungen (nur Wiegescheine)“ = Import-Container „TIM_WIEGESCHEIN“
 - ⚠ In der Interface ist nur der Import-Container TIM_WIEGESCHEIN angelegt; entsprechend können nur die Wiegescheine der Verwiegungen importiert werden.
- „Verwiegungen (nur Kostenstellenanteile)“ = Import-Container „KOSTENSTELLENANTEIL“
 - ⚠ In der Interface ist nur der Import-Container KOSTENSTELLENANTEIL angelegt; entsprechend können nur die Kostenstellenanteile der Verwiegungen importiert werden.

6.2.1. Wurzel-Element <interface>

<interface>		Default	Pflicht	Anzahl
name	Name, entspricht dem Bezeichner		Ja	
container			Nein	
transaction	Transaktionsklammer auf Interface	true	Nein	
stop-on-error	Schnittstelle bricht bei Fehler in einer Zeile ab	true	Nein	
<file-definition>	verweist auf file-definition		Ja	1
<import-schema>	verweist auf Import-Schema		Ja	1
<multi-mapping>	verweist auf Schlüsseldefinition		Nein	0...n

Die Attribute *transaction* und *stop-on-error* sind im Kapitel Transaktionsmanagement (siehe [Kapitel 7](#)) beschrieben.

6.2.2. Element <file-definition>

<file-definition>		Default	Pflicht	Anzahl
href	Pfad und Name der -file-definition.xml		ja	

6.2.3. Element <import-schema>

<import-schema>		Default	Pflicht	Anzahl
typeFormattingRule set	gibt an, welches Format für Datums- und Zahlenangaben Anwendung finden soll: „wme.fact“ entsprechend dem Format der Schnittstelle (tt.mm.yyyy; Dezimaltrenner Komma) „W3C-Schema“ entsprechend dem Format der Meldeschnittstelle (yyyy-mm-tt; Dezimaltrenner Punkt)	wme.fact	nein	
ref	Verweist auf ein Import-Schema. Die Angabe ist bei Import von XML-Daten Pflicht, in diesem Fall ist als Wert „os“ einzugeben.		(ja)	
<import-container>	verweist auf Import-Container		nein	0...n

Das Element `<import-schema>` klammert zunächst die `<import-container>` - Elemente. Im Falle eines Imports von XML - Daten kommt diesem Element mehr Bedeutung zu. Mit dem Attribut „ref“ muss hier ein Schema - Name angegeben werden, welchem das zu importierende XML-File entsprechen muss. Momentan ist hier nur „os“ als Schema-Name zugelassen. Die entsprechende Schema-Datei ist `<wme.fact>\Config\Transfer\schema\os.xsd`. Entspricht die zu importierende XML-Datei nicht diesem Schema, so bricht der Import mit einem Fehler ab. In diesem Fall wird die Möglichkeit gegeben, die nicht schemakonforme XML-Datei während des Imports per XSL-Transformation entsprechend umzuformen (mehr dazu in [Kapitel 6.4.4.](#)).

6.2.4. Element `<import-container>`

Dies ist die zentrale Stelle, um die Datenbanktabellen mit Werten der Importdatei zu belegen.

<code><import-container></code>		Default	Pflicht	Anzahl
ref	Name des Import - Containers		ja	
field-type	XML - Element bei XML-Import, welches den Datensatz enthält		nein	
group	Gruppierung der Sätze dieses Import - Containers auf einen vorhandenen Satz der Zwischentabellen	false	nein	
<code><import-container></code>	Container für Details			0...n
<code><variable></code>	Variablenbelegung			1...n
<code><master></code>	Verknüpfung mit Master			0...1

Das Attribut **field-type** spielt nur beim XML - Import eine Rolle; hier wird angegeben mit welchem XML-Element dieser Container gefüllt wird.

Das Attribut **group** bewirkt, dass zunächst nach bereits vorhandenen Objekten dieser Entität gesucht wird, und dann das erste Element mit dem momentanen Datensatz aktualisiert wird (siehe [Kapitel 8](#)).

Das Kind - Element **import-container** bewirkt, dass alle Entitäten des Kind - Containers automatisch als „Detail“ aufgehängt werden. Das heißt, dass allein die Schachtelung der Import-Container-Elemente die Master-Detail-Verknüpfung bewirkt.

Folgende Import - Container sind möglich:

- TIM_VERWIEGUNG
- TIM_WIEGESCHEIN
- KOSTENSTELLENANTEIL

Dabei ist TIM_VERWIEGUNG der äußere Master-Container für die Details-Container TIM_WIEGESCHEIN und KOSTENSTELLENANTEIL.

Der nachfolgende Ausschnitt aus der BA31-interface.xml soll die Struktur nochmals verdeutlichen:

```
<import-container ref="TIM_VERWIEGUNG">
```

```

...
    <import-container ref="TIM_WIEGESCHEIN">
...
    </import-container>
    <import-container ref= »KOSTENSTELLENANTEIL »>
...
    </import-container>
</import-container>

```

6.2.4.1. Element <variable>

<variable>		Default	Pflicht	Anzahl
ref	Verweis auf eine Variable in import-container.xml, d.h. der Zielvariablen ^{wm} e.fact		ja	
field	Verweis auf ein Feld in der Importdatei, d.h. auf eine Quellvariable		nein	
default	ein fester Wert		nein	
group	Attribut wird zur select-Klausel hinzugefügt	false	nein	
format	Formatangabe z.B. ddmmyy		nein	
multi-mapping	Referenz auf target-Feld einer Schlüsseldefinition		nein	
<get-id-from>	Fremdschlüsselreferenz			0...1
<filter>	Filter zur Bearbeitung von Werten			0...n

Das Attribut **ref** verweist auf eine Variable aus dem übergeordneten Import-Container (bitte beachten: hier muss Groß-/Kleinschreibung stimmen). Bereits beim Parsen der Dateien werden die Attribute überprüft.

Attribut **field** gibt den Spaltennamen, bzw. das XML-Element der Import-Datei an. Der Spaltenname ist entweder durch –file-definition.xml definiert oder bereits vorhanden (CSV mit Überschrift, bzw. XML). Mit dem Attribut **default** wird die Variable mit einem festen Wert belegt. Die Angabe von field und default ist gemeinsam möglich.

Attribut **group** besagt, dass nach dieser Variable zunächst in der (TIM_-)Tabelle gesucht wird. Das heißt, dass alle Variablen, die das Attribut group gesetzt haben, die WHERE-Klausel der Suche bilden. Wird ein Satz gefunden, wird dieser aktualisiert. I. d. R. werden die group-Attribute auf den Feldern des Unique-Key liegen.

Attribut **format** erlaubt die Angabe von speziellen Formaten, z.B. Datumsformaten. Im [Anhang 11.5](#) werden die verschiedenen Formate näher erläutert.

Das Attribut **multi-mapping** referenziert ein „target-Feld“ einer Schlüsseldefinition (siehe [Kapitel 6.3](#))

6.2.4.2. Element <get-id-from>

<get-id-from> wird intern für die Auflösung von Zielvariablen benötigt (siehe nachfolgendes Beispiel).

Ist eine Variable der ^{wm}e.fact Zieltabelle ein Fremdschlüsselfeld, so muss aus der Fremdtabelle der Foreign-Key ermittelt werden.

<get-id-from>		Default	Pflicht	Anzahl
container	Name des Fremdschlüssel-Containers (nicht zu verwechseln mit Import-Container)		ja	
<variable>	Variablenbelegung			1...n

1.1.16 Beispiel

```
<variable ref="ARTIKEL">
<get-id-from container="ARTIKEL">
<variable ref="v_Artikelnummer" field="FRAKT_NR"/>
</get-id-from>
</variable>
```

Alle Variablen des get-id-from-containers bilden die Where-Klausel auf der Suche nach dem Fremdschlüssel.

Das obige Beispiel liest sich folgendermaßen:

Weise der Variablen ARTIKEL den technischen Ident desjenigen Datensatzes zu, dessen Wert für die Spalte v_Artikelnummer im Container ARTIKEL (= Tabelle TST_ARTIKEL) dem Wert des Feldes FRAKT_NR entspricht.

oder in SQL:

„select PN_ARTIKELID from TST_ARTIKEL where v_Artikelnummer = FRAKT_NR“

6.2.4.3. Element <filter>

Filter ermöglichen es, die zu importierenden Daten unmittelbar vor der Zuweisung zu Variablen bzw. die zu exportierenden Daten unmittelbar nach dem Auslesen aus der Datenbank zu transformieren. Für die Anwendung von Filtern spielt es keine Rolle, ob für das jeweilige Datenfeld ein Format definiert ist.

Gegenwärtig stehen zwei Filter zur Verfügung:

- Ein Substring Filter, mit dem es möglich ist Teile aus einer Zeichenkette zu extrahieren
- Ein Regular Expression Filter, der mit Hilfe von sog. Regulären Ausdrücken eine nahezu beliebige Manipulation von Zeichenketten ermöglicht

Die Konfiguration und das „Einschalten“ eines Filters für ein bestimmtes Feld erfolgt in den entsprechenden Schnittstellendefinitionen (-interface.xml) mit Hilfe des Elements <filter>.

Dieses Element kann in Export-Definitionen als Kind von <field>, in den Import-Definitionen als Kind von <variable> erscheinen.

⚠ Die folgenden Erläuterungen beziehen sich i.d.R. nur auf den Import, da das Vorgehen für den Export von Daten im Hinblick auf Filtermöglichkeiten vollkommen identisch ist.

⚠ Für die gesamte Transformation wurde eine externe RegeX - Bibliothek in die Waageschnittstelle eingebunden. Weiterführende Literatur über www.jakarta.apache.org/oro/

<filter>		Default	Pflicht	Anzahl
<apply-if>	Bedingung nach der der Filter greifen soll		nein	0..n
<substring>			Eins	
<regexp>			von beiden	

Die einfachste mögliche Filterdefinition sieht folgendermaßen aus:

Beispiel:

```
<filter>
    <substring from="0" length="5"/>
</filter>
```

Hier handelt es sich um einen Filter mit einer einzigen Transformation, die für das gefilterte Feld immer ausgeführt wird: „Importiere vom referenzierten Feld nur die ersten fünf Zeichen.“

Allerdings kann ein Filter auch deutlich komplexer sein:

- Ein Filter kann mehrere Transformationen enthalten
- Gruppen von Transformationen können in Abhängigkeit von einer oder mehreren Bedingungen durchgeführt werden

6.2.4.4. Element <apply-if>

<apply-if> dient dazu, in Abhängigkeit von einer Bedingung zu entscheiden, ob eine Transformation ausgeführt werden soll. Hierzu wird der zu importierende/exportierende Wert gegen einen Regulären Ausdruck geprüft. Wenn der Wert auf den Ausdruck „passt“, dann ist das Ergebnis der Bedingung TRUE und die Transformation wird ausgeführt, sonst FALSE.

<apply-if>		Default	Pflicht	Anzahl
pattern	Der reguläre Ausdruck, gegen den der Wert geprüft wird		Ja	
ignore-case	Groß-/Kleinschreibung ignorieren	false	Nein	
full-match	Das Muster muss den Wert vollständig abdecken	false	Nein	
invert	Die nachfolgenden Filter werden ausgeführt, wenn der Wert nicht auf pattern passt (d.h. das Ergebnis wird logisch negiert).	false	Nein	

Das Attribut **pattern** enthält den Regulären Ausdruck, gegen den der Wert geprüft wird.

Das Attribut **ignore-case** gibt an, ob pattern Groß/Kleinschreibung im Wert ignorieren soll.

Das Attribut **full-match** gibt an, ob pattern den gesamten Wert abdecken muss, oder ob bereits Teile des Wertes ausreichen.

6.2.4.5. Element <substring>

Das <substring> - Element definiert eine Transformation, die aus einer Eingabezeichenkette eine Teilzeichenkette (einen Substring) extrahiert und diesen zur Verarbeitung weiterreicht. Der gleiche Effekt könnte zwar auch mit einer entsprechenden <regex> - Transformation erreicht werden, jedoch ist die Verwendung von <substring> ggf. bequemer und leichter zu lesen.

<substring>		Default	Pflicht	Anzahl
from	Position des ersten auszugebenden Zeichens (Zählung beginnt bei 0!)		Ja	
length	Anzahl der auszugebenden Zeichen	bis zum Ende / Anfang	Nein	
backwards	Zählung von hinten beginnen. D.h. from=0, length=4 gibt die letzten vier Zeichen aus	false	Nein	

6.2.4.6. Element <regex>

Das <regex> - Element definiert eine Transformation, die eine Eingabezeichenkette mit Hilfe von Regulären Ausdrücken und eines Suchen/Ersetzen Mechanismus in eine Ausgabezeichenkette zur Weiterverarbeitung transformiert. Die Möglichkeiten dabei sind überaus vielfältig.

<regex>		Default	Pflicht	Anzahl
pattern	Das Suchmuster (ein Regulärer Ausdruck)		Ja	
replacement	Das Ersetzungsmuster		Ja	
ignore-case	Groß-/Kleinschreibung in der Suche ignorieren	false	Nein	
replace-once	Die Ersetzung wird nur für den ersten Treffer des Musters durchgeführt	false	Nein	

Einige Beispiele zu Filtern befinden sich in der Datei <wme.fact>\Schnittstelle\WsBA31Beispiel\BA31-interface.xml

6.2.4.7. Element <master>

Mit dem Element <master>, der i.d.R. als letzter Container in der -interface.xml hinterlegt wird, erfolgt die Verknüpfung eines Satzes zu einem bereits im temporären Bereich (TIM_) vorhandenen Master.

<master>		Default	Pflicht	Anzahl
Container	Name des Master-Containers		ja	
<follow-relationship-forward>	Auflösung erfolgt über einen anderen Container	nur eins von beiden		
<variable>	Variablenbelegung			1...n

Die Elemente <follow-relationship-forward> und <variable> definieren jeweils die Suche nach dem Master. Es ist nur jeweils eine Variante erlaubt. Mit dem Element <variable> wird im angegebenen Master-Container nach den Sätzen gesucht, die sich über die Variablen - Belegung ergeben.

Beispiel zu Verwendung von <variable>:

```
<master container="TIM_VERWIEGUNG">
<variable ref="AUFTRAGSNUMMER" field="AuftragsnummerFrei">
</master>
```

Das Beispiel liest sich wie folgt:

„Suche in TIM_VERWIEGUNG den Datensatz, dessen AUFTRAGSNUMMER gleich AuftragsnummerFrei ist“

Diese Funktionalität wird z.B. dann benötigt, wenn die zu importierenden Sätze nur Wiegescheindaten enthalten und mehrere Wiegescheine an einen bereits in der Zwischentabelle vorhandenen Verwiegungsmaster angehängen werden sollen. In unserem Beispiel würden die Wiegescheine aus allen Sätzen mit der gleichen freien Auftragsnummer unter einem Verwiegungsmaster angehängen.

Würde im vorliegenden Beispiel die Referenz nicht über den Master gesetzt, so wird der erste angehängene Wiegeschein durch den zweiten Wiegeschein überschrieben.

6.2.4.8. <follow-relationship-forward>:

<follow-relationship-forward>		Default	Pflicht	Anzahl
container	Name des Ausgangs - Containers		ja	
<variable>	Variablenbelegung			1...n

Durch die Anweisung wird folgendes zum Ausdruck gebracht:

„Hänge den neuen Satz unter dem Verwiegungsmaster an, unter welchem schon ein bestimmter anderer Satz zugeordnet wurde.“

1.1.17 Beispiel:

```
<import-container ref="TIM_WIEGESCHEIN">
...
<master container="TIM_VERWIEGUNG">
<follow-relationship-forward container="TIM_WIEGESCHEIN">
<variable ref="WIEGESCHEIN_NR" field="12345">
</follow-relationship-forward>
</master>
</import-container>
```

Liest sich wie folgt:

„Hänge den Wiegeschein unter der Verwiegung an, unter welcher der Wiegeschein mit der Wiegescheinnummer 12345 bereits zugeordnet ist.“

Im Gegensatz zum Ansatz `<variable>` (Suche direkt nach einem Verwiegungsmaster, der den Wert der Quellvariable enthält) wird über `<follow-relationship-forward>` über die Details nach einem Verwiegungsmaster gesucht.

Wichtig ist bei beiden Varianten, dass mit dem `<master>` - Element nach einem bereits vorhandenen Datensatz gesucht wird. Ist dieser nicht vorhanden, so kommt es zu einem Zeilenfehler; der eigentliche Satz kann nicht importiert werden.

6.2.4.9. Import von Master und Detail

Will man in einer Schnittstelle Master *und* Detail importieren, z.B. Verwiegungen und Wiegescheine, so definiert man dies nicht über das `<master>` - Element, sondern über geschachtelte Import-Container:

```
<import-container ref="TIM_VERWIEGUNG">
  ...
  <import-container ref="TIM_WIEGESCHEIN">
    ...
  </import-container>
</import-container>
```

Die Importdaten liegen in einem solchen Fall in einer einzigen Datei vor. Jeder Datensatz enthält somit sowohl die Masterdaten als auch die jeweiligen Wiegescheindaten.

Sollen hier mehrere Sätze auf einen Verwiegungsmaster verdichtet werden, so findet das Element `<group>` Anwendung (siehe [Kapitel 8](#))

6.2.4.10. Element `<multi-mapping>`

Mit diesem Element werden zunächst nur Schlüssel - Definitionen „inkludiert“. Es sind beliebig viele sind erlaubt, wobei auf die Eindeutigkeit des Namens zu achten ist.

<code><multi-mapping></code>		Default	Pflicht	Anzahl
name	Vergabe eines Alias		ja	
href	Pfadangabe der Schlüsseldefinition			
<source>	Verknüpft das Quell-Feld einer Schlüsseldefinition mit einem Eingabe-Feld. Die Attribute sind dieselben wie bei <code><variable></code>			1...n

Beispiel:

```
<multi-mapping name="befoerderung-alias" href="Multimapping\befoerderung-mm.xml">
  <source ref="BEFOERD_ALT" field="BEFOERD"/>
</multi-mapping>
```

Die Verwendung einer Schlüsseldefinition in einer Variablen erfolgt folgendermaßen:

```
<variable ref="n_Kennnummer" multi-mapping="befoerderung-alias/BEFOERDARTEN_KENNR"/>
```

Die Variable `n_Kennnummer` wird mit demjenigen Target-Feld (`BEFOERDARTEN_KENNR`) der Schlüsseldefinition `Multimapping\befoerderung-mm.xml` belegt, dessen Source-Feld (`BEFOERD_ALT`) dem Wert des Import-Feldes (`BEFOERD`) entspricht.

6.3. Multi-Mapping

Beschreibt eine Umschlüsselung von „Quell-Werten“ zu „Ziel-Werten“. Neben der Definition ist auch der eigentliche Inhalt vorhanden.

Format:

```
<multi-mapping
  xmlns="http://www.dsd-ag.de/wme-fact/transfer/multi-mapping"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/multi-mapping
D:\wme.fact\Config\Transfer\schema\multi-mapping.xsd"
  name="String">
  <definition>
    <source/>
    <target/>
  </definition>
  <content/>
</multi-mapping>
```

6.3.1. Wurzel-Element <multi-mapping>

<multi-mapping>		Default	Pflicht	Anzahl
name	Name		Ja	
<definition>	Definitionsteil		Ja	1
<content>	Inhaltsteil		Ja	1

6.3.2. Element <definiton>

<definition>		Default	Pflicht	Anzahl
<source>	Quell-Feld		Ja	1..n
<target>	Ziel-Feld		Ja	1..n

<source>		Default	Pflicht	Anzahl
name	Name des Source-Elements		ja	
operator	Werte: EQUALS oder STARTS-WITH oder CONTAINS	EQUAL S	nein	

<target>		Default	Pflicht	Anzahl
name	Name des Target-Elementes		ja	
operator	Werte: EQUALS oder STARTS-WITH oder CONTAINS	EQUAL S	nein	

6.3.3. Element <content>

<content>		Default	Pflicht	Anzahl
<record>	ein einzelner Umschlüsselungssatz		Nein	0..n
<default>	Der „Fallback-Satz“		Nein	0..1

<record>		Default	Pflicht	Anzahl
<source>	Name wie im Definitionsteil			1..n
<target>	Name wie im Definitionsteil			1..n

<default>		Default	Pflicht	Anzahl
<record>	enthält einzelnes target			1

Beispiel:

```

<multi-mapping
xmlns="http://www.dsd-ag.de/wme-fact/transfer/multi-mapping"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/multi-mapping
../Config/Transfer/schema/multi-mapping.xsd"
name="retoure-mm">
<definition>
<source name="RETOUR_KZ_ALT"/>
<target name="RETOURE_KENNR"/>
</definition>

<content>
<record>
<source name="RETOUR_KZ_ALT">F</source>
<target name="RETOURE_KENNR">1</target>
</record>
<record>
<source name="RETOUR_KZ_ALT">T</source>
<target name="RETOURE_KENNR">2</target>
</record>
<default>
<target name="RETOURE_KENNR">1</target>
</default>
</content>
</multi-mapping>

```

6.3.4. Besonderes***-Vergleich**

Es ist möglich mit * zu vergleichen, d.h. der Source-Datensatz darf dann einen beliebigen Inhalt haben.

Beispiel:


```

<multi-mapping name="mm_tourzugebiet">
<definition>
<source name="mms_tour"></source>
<target name="mmf_gebiet"></target>
</definition>
<content>
<record>
<source name="mms_tour">*</source>
<target name="mmf_gebiet">Bayern</target>
</record>
</content>
</multi-mapping>

```

Ein ausführlicheres Beispiel einer multi-mapping.xml befindet sich in [Kapitel 9](#).

6.3.5. Eindeutigkeiten

 Ein Multi-Mapping kann im Import nur genutzt werden, wenn der Wert der Source-Variable(n) jeweils eindeutig auf einen Wert der Target-Variablen verweist.

Im Import können dabei unterschiedliche Source-Werte auf den gleichen Target-Wert gelegt werden. Im Export ist die Bedingung genau umgekehrt.

Soll ein Multi-Mapping sowohl im Import als auch im Export Verwendung finden, so sind Source- und Target-Werte nur in einem 1:1-Verhältnis zulässig.

6.4. -file-definition.xml

Beschreibt das Datenformat und somit die Felder der Import-Datei. Für den Fall, dass die Import-/Export-Datei keine Spaltenüberschriften enthält, können hier Aliase für die Felder eingeführt werden. Mit diesen Aliasen kann in der Schnittstellen-Definition (-interface.xml) dann gearbeitet werden.

Eine File - Definition kann i.d.R. so gestaltet werden, dass sie sowohl für den Import, als auch für den Export dient.

1.1.18 Template:

<file-definition

```
xmlns="http://www.dsd-ag.de/wme-fact/transfer/file-definition"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/file-definition
file:///D:/wme.fact/Config/Transfer/schema/file-definition.xsd"
name="Formatdefinition Schnittstelle <Bezeichnung>"
format="txt">
<xml/>
<fixed>
  <field name="__Platzhalter__" offset="0"/>
  </fixed>
<txt header="false" separator="|" flavour="microsoft">
<field name="__Platzhalter__" column="1"/>
</txt>
```

</file-definition>

6.4.1. Wurzel-Element <file-definition>

<file-definition>		Default	Pflicht	Anzahl
format	Werte: txt, xml, fixed (fixed nur bei Import)		ja	
name	Name		ja	
<txt>	je nach Wert von Attribut „format“ wird auf das entsprechende Element zugegriffen		ja	1
<xml>			ja	1
<fixed>			ja	1

6.4.2. Element <txt>

<txt>		Default	Pflicht	Anzahl
separator	der Spaltentrenner	TAB	nein	
header	sind Überschriften vorhanden (true / false)	true	nein	
flavour	Vorgabe für das Escapen des Feldtrennzeichens innerhalb eines Wertes. <u>microsoft</u> escaped durch Verdoppelung des Feldtrenners, <u>standard</u> escaped durch \	microsoft	nein	
<field>	zur Vergabe eines Feldnamens		nein	0...n

Als Separator kann man eine beliebige Zeichenkette definiert werden. Das Tabulatorzeichen wird durch „TAB“ repräsentiert.

Mit Hilfe des <field> - Elementes kann für eine Spalte ein Alias vergeben werden, welcher dann in einer Variablen - Belegung mit dem Attribut *field* benutzt werden kann. Sind Spaltenüberschriften vorhanden, so können diese hiermit umbenannt werden. Sind keine Spaltenüberschriften vorhanden, sollte die Möglichkeit genutzt werden, die Spalten zu benamen. In der Schnittstellen-Definition kann auch mit der Angabe der Spaltenposition gearbeitet werden (field="1").

6.4.2.1. Element <field>

<field>		Default	Pflicht	Anzahl
name	Vergabe eines Alias		ja	
column	Position der Spalte, bei 1 beginnend, oder originale Spaltenüberschrift		ja	

6.4.3. Element <fixed>

Definiert für das Format „Feste Satzlänge“ die logischen Feldnamen.

<fixed>		Default	Pflicht	Anzahl
offset-zero	Position der „nullten“ Spalte	0	nein	
<field>	Zur Vergabe eines Feldnamens		ja	0...n

Die <field> - Elemente sind hier zwingend notwendig, da ansonsten der Interface-Definition keine Feldnamen zur Verfügung stehen.

6.4.3.1. Element <field>

<field>		Default	Pflicht	Anzahl
name	Vergabe eines Alias		ja	
length	Länge der Spalte		Min. eins von beiden	
offset	Startposition der Spalte			

Fehlt das Attribut *offset*, ergibt sich der Beginn einer Spalte aus der Länge der Vorgänger-Spalte.

Fehlt die Angabe von *length*, ergibt sich die Länge aus dem Offset der Nachfolger-Spalte.

6.4.4. Element <xml>

Für das Format XML sind keinerlei Feldnamen zu definieren. Für den XML-Import sind feste Schemata definiert und somit auch die Feldnamen bereits vorgegeben. Um auch Fremdformate zu importieren, besteht die Möglichkeit eine XSL-Transformation vorzuschalten, welche die Importdatei in eine gültige^{wm}e.fact Importdatei umwandelt.

<xml>		Default	Pflicht	Anzahl
<transformer>	definiert XSL-Transformation		Nein	0...1

Entspricht die XML-Importdatei nicht dem angegebenen Schema (siehe 5.2.3) so muss das Element <transformer> angegeben und mit einer xsl-Datei verbunden werden (Attribut **href**).

6.4.4.1. Element <transformer>

<transformer>		Default	Pflicht	Anzahl
href	Pfad der XSL-Datei		Ja	

Das Attribut **href** gibt die XSL-Datei an, die die nicht schemakonforme XML-Importdatei in eine schemakonforme XML-Datei umwandelt. Während des Imports wird die Transformation on-the-fly durchgeführt. Sie hat keinen Einfluss auf die physikalische Importdatei.

6.4.5. Mehr zum TXT-Format

Ein Spaltentrenner muss in jedem TXT-Format vorhanden sein. Somit sind letztlich die Parameter „Spaltenüberschriften Ja/Nein“ und „Felddefinition* vorhanden Ja/Nein“ relevant. Hieraus resultieren 4 Fallgruppen:

	Header	Kein Header
Felddefinition vorhanden	Beispiel: <code><field name="TARA_ZEIT" column="TZEIT"/></code> Das logische Feld „TARA_ZEIT“ wird mit dem Wert aus der Spalte „TZEIT“ belegt. Das log. Feld kann in der -interface.xml folgendermaßen verwendet werden: <code><variable ref="TARAZEIT" field="TARA_ZEIT"/></code>	Beispiel: <code><field name="TARA_ZEIT" column="3"/></code> Das logische Feld „ TARA_ZEIT „ wird mit dem Wert der 3.Spalte belegt. Das log. Feld kann in der -interface.xml folgendermaßen verwendet werden: <code><variable ref="TARAZEIT" field="TARA_ZEIT"/></code>
Keine Felddefinition	Als Feldnamen können in der interface.xml direkt die Spaltennamen des TXT-Files verwendet werden. Bsp: <code><variable ref="TARAZEIT" field="TZEIT"/></code>	Weder log. Feldnamen sind definiert, noch enthält das TXT-File Spaltenüberschriften. In der interface.xml stehen somit nur die Spaltenpositionen zur Verfügung. Bsp.: <code><variable ref="TARAZEIT" field="3"/></code>

- Felddefinition bezeichnet hier die field - Elemente in der -file-definition.xml.

7 Transaktions-Management

Das Transaktionsmanagement steuert das Verhalten beim Import von fehlerhaften Sätzen während des Importvorgangs. Wann soll eine Transaktion rückgesetzt werden?

Es gibt mehrere Möglichkeiten die Transaktions-Klammer für den Import zu setzen, nämlich auf der Ebene

- eines Satzes (Master im Falle von XML)
- einer Schnittstelle
- der Routine

Die Klammer kann im XML mit dem Attribut „transaction“ gesetzt werden.

Des Weiteren werden 2 Arten von Fehlern unterschieden:

- Fehler in der Schnittstelle
- Fehler im Datensatz

Für die Behandlung eines Satzfehlers stehen wiederum 2 Möglichkeiten zur Auswahl:

- Den Import dieser Schnittstelle (Verwiegung, Wiegeschein oder Kostenstellenanteil) abbrechen
- Ausgabe des Fehlers im Log und Fortfahren

Ist das Attribut transaction = true gesetzt, so kann über das Attribut „stop-on-error“ in der Schnittstellendefinition das Verhalten gesteuert werden. Somit ergibt sich folgende Matrix:

	Exception in Schnittstelle	Fehler/Exception im Datensatz	
TA-Ebene		Stop-on-error = „true“	Stop-on-error = „false“
Master /Satz	Import wird abgebrochen, kein Rollback der vorher gelaufenen Jobs (denn diese sind ja bereits committed)	Rollback des aktuellen Masters; Import stoppt; Vorher importierte Master bleiben importiert	Rollback des aktuellen Masters plus Details; Import läuft weiter
Interface	Import wird abgebrochen, kein Rollback der vorher gelaufenen Jobs (denn diese sind ja bereits committed)	Rollback des aktuellen Jobs und somit Abbruch des Imports (kein Rollback der vorherigen Jobs)	Ausgabe des fehlerhaften Satzes im Log, Job läuft weiter
Import-Routine	Rollback des gesamten Imports	Rollback des gesamten Imports	Ausgabe des fehlerhaften Satzes im Log, Job läuft weiter

Beispiel für ‚Exception in Schnittstelle‘:

- Datendatei kann nicht geöffnet werden

Beispiel für ‚Exception im Datensatz‘:

- In der Interface ist eine Zielvariable einer nicht bekannten Quellvariable zugeordnet

7.1. Beispiel

Eine Routine besteht aus 2 Schnittstellen A und B. In A werden 3 Sätze fehlerlos importiert, in B liegen 5 Sätze vor, der 2. Satz macht aber Probleme.

Je nach Konfiguration (Lage in der Matrix) variiert die Anzahl der nun tatsächlich importierten Sätze.

Folgende Matrix skizziert die Ausgaben des Logfiles.

Beispiel für „Exception in Schnittstelle“:

Importdatei, mit welcher B gestartet werden soll, ist fehlerhaft.

Beispiel für „Exception im Satz“:

Der 2.Satz aus B enthält eine nicht vorhandene Kennnummer.

	Exception in Schnittstelle	Fehler/Exception im Datensatz	
TA-Ebene		Stop-on-error = „true“	Stop-on-error = „false“
Master /Satz	A: 3 committed 0 fehlgeschlagen B: 0 committed 0 fehlgeschlagen	A: 3 committed 0 fehlgeschlagen B: 1 committed 1 fehlgeschlagen	A: 3 committed 0 fehlgeschlagen B: 4 committed 1 fehlgeschlagen
	-> 3 importiert DSD7029	4 importiert DSD7029	7 importiert DSD7050
Interface (B-interface.xml)	A: 3 committed 0 fehlgeschlagen B: 0 committed 0 fehlgeschlagen	A: 3 committed 0 fehlgeschlagen B: 0 committed 1 fehlgeschlagen	A: 3 committed 0 fehlgeschlagen B: 4 committed 1 fehlgeschlagen
	3 importiert DSD7029	3 importiert DSD7029	7 importiert DSD7050
Import-Routine	A: 3 noch nicht committed 0 fehlgeschlagen B: 0 noch nicht committed 0 fehlgeschlagen	A: 3 noch nicht committed 0 fehlgeschlagen B: 1 noch nicht committed 1 fehlgeschlagen	A: 3 noch nicht committed 0 fehlgeschlagen B: 4 noch nicht committed 1 fehlgeschlagen
	0 importiert DSD7028	0 importiert DSD7028	7 importiert DSD7050

7.2. Fehlermeldungen

Aus der obigen „Fehlermatrix“ ergeben sich automatisch die diversen Fehlermeldungen, die nach dem Import angezeigt werden. 3 Haupt-Meldungen werden unterschieden:

- DSD7029=“Der Datenimport ist fehlerhaft. Bitte prüfen Sie die Protokolldatei“:
Immer dann, wenn nicht alle Sätze importiert wurden
- DSD7028=“Der Datenimport ist fehlgeschlagen. Der Urzustand ist wieder hergestellt“:
Hier wurde nach einem Fehler ein vollständiger Rollback vorgenommen (siehe letzte Zeile der obigen Abbildung)
- DSD7050=“Der Datenimport ist erfolgreich verlaufen, aber {0} Datensatz/Datensätze sind nicht mit übernommen worden“:
Immer dann, wenn ein Fehler auftauchte, aber trotzdem fortgefahren wurde, also somit nur bei *stop-on-error=false* (siehe letzte Spalte der obigen Abbildung).

Sofern es möglich ist, enthalten die Meldungen weitere Sub-Meldungen, um dem Benutzer so viele Infos wie möglich zu geben.

- ⚠ Da auf allen 3 Ebenen das Attribut „stop-on-error“ gesetzt werden kann, ist seitens des Anwenders Sorge dafür zu tragen, dass diese in allen drei Fällen gleich gesetzt sind.

8 Ausführungen zur Group-Funktionalität

Zwei unterschiedliche Anforderungen für eine Gruppierung sind zu unterscheiden:

- Innerhalb eines Importvorgangs sollen Details auf einen Master gruppiert werden
- Details eines Importvorgangs sollen auf bereits importierte Mastersätze gruppiert werden

8.1. Group innerhalb einer Importdatei

In diesem Fall sind alle die Elemente im Master mit dem Attribut group = "true" zu versehen, die die Eindeutigkeit des Masters bestimmen. Die Detailinformationen werden dann dem Master zugeordnet.

1.1.19 Beispiel:

Gruppierung aller Wiegescheine auf den Verwiegungsmaster mit derselben Auftragsnummer und demselben Garantiegeber.

```
<variable ref="AUFTRAGSNUMMER" field="WIEGE_NR" group="true"/>
```

```
<variable ref="GARANTIEGEBER" group="true">
```

```
    <get-id-from ...>
```

```
    ...
```

```
</variable>
```

8.2. Group auf einen bereits importierten Master

Anders gelagert ist der Fall, wenn der Master bereits importiert ist (in dem temporären Bereich) und die Details angehängen werden sollen. In diesem Falle ist im interface.xml des Details am Ende des Import-Containers der Verweis auf den eindeutigen Master zu hinterlegen. Damit erfolgt automatisch die gewünschte Gruppierung. Das Group-Attribut wird nicht mehr gesetzt. Siehe dazu auch [Kapitel 6.2.4.7.](#) und [Kapitel 6.2.4.8.](#)

1.1.20 Beispiel:

```
...
```

```
    <master container="TIM_VERWIEGUNG">
```

```
<variable ref="AUFTRAGSNUMMER" field="V_AUFTRAGSNR" />
```

```
    </master>
```

```
</import-container>
```

```
    </import-schema>
```

```
</interface>
```

9 Ausführungen zum Multi - Mapping

Sollen mehr als eine Source- oder Target - Variable Verwendung finden, so stellt sich die Frage nach der Einbindung in der interface.xml. Im nachfolgenden Beispiel sollen die Source-Variablen „Belegart“ und „beliebte Anlage“ auf die Target-Variablen „Buchungsart“, „Wareneingang“ und „Senderwieschein“ gemappt werden.

Hierzu sind die Bearbeitungsschritte des Mappings zu beachten:

- Im Ablauf des Imports werden zunächst die hinterlegten Multi-Mappings komplett geladen
- Die Zuweisungen der durch in einem Importsatz gefüllten Variablenwerte lösen als erstes die Source-Variablen des Mappings auf und halten alle gefundenen Target-Werte vor
- Erst dann erfolgt die Zuordnung der Variablenwerte des Importsatzes zu den Variablen der Zieltabellen
- Mit dem Einlesen eines neuen Importsatzes werden die Source-Variablen neu aufgelöst

Damit ist es ausreichend, in der Zuordnungsvorschrift der interface.xml jeweils nur die relevanten Stellen auf die Zielvariable zu verweisen. Im nachfolgenden Beispiel werden im Import 5 Source-Variablen auf 7 Target-Variablen verknüpft.

Beispiel Multi-Mapping

```

1  <?xml version="1.0" encoding="ISO-8859-1"?>
2
3  <multi-mapping
4    xmlns="http://www.dsd-ag.de/wme-fact/transfer/multi-mapping"
5    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
6    xsi:schemaLocation="http://www.dsd-ag.de/wme-fact/transfer/multi-mapping ../../Config/Transfer/schema/multi-mapping.xsd"
7    name="buchungsart-mm">
8
9    <definition>
10     <source name="BA_NR_ALT"/>
11     <source name="ANL_NR_BEL_ALT"/>
12     <source name="ANL_NR_HER_ALT"/>
13     <source name="DUPLIZIER_ALT"/>
14     <source name="STO_GRD_ALT"/>
15
16     <target name="BUCHUNGSART_KENNR"/>
17     <target name="WARENEINGANG"/>
18     <target name="SENDERWIEGESCHEIN"/>
19     <target name="VERTRAGSPARTNER_KTONR"/>
20     <target name="VERTRAGSPARTNER_KTOKLASSE"/>
21     <target name="GG_KTONR"/>
22     <target name="GG_KTOKLASSE"/>
23   </definition>
24
25   <content>
26
27     <record>
28       <source name="BA_NR_ALT">31</source>
29       <source name="ANL_NR_BEL_ALT">11111</source>
30       <source name="ANL_NR_HER_ALT">0</source>
31       <source name="DUPLIZIER_ALT">F</source>
32       <source name="STO_GRD_ALT">00</source>
33
34       <target name="BUCHUNGSART_KENNR">100</target>
35       <target name="WARENEINGANG">T</target>
36       <target name="SENDERWIEGESCHEIN">F</target>
37       <target name="VERTRAGSPARTNER_KTONR">48114</target>
38       <target name="VERTRAGSPARTNER_KTOKLASSE">Anlieferer</target>
39       <target name="GG_KTONR">48117</target>
40       <target name="GG_KTOKLASSE">Garantiegeber</target>
41     </record>
42
43   </content>
44 </multi-mapping>

```

Abbildung 9-1: Multi-Mapping buchungsart-mm.xml

Beispiel für die Verknüpfung des Multi-Mappings in BA31-interface.xml

```
...
```

```
<variable ref="BUCHUNGSART">
```

```
<get-id-from container="BUCHUNGSARTEN">
```

```
<variable ref="n_Kennnummer" multi-mapping="buchungsart-alias  
/BUCHUNGSART_KENNR"/>
```

```
</get-id-from>
```

```
</variable>
```

```
<variable ref="VERTRAGSPARTNER">
```

```
<get-id-from container="PERSONENKONTO">
```

```
<variable ref="v_KontoNR" multi-mapping="buchungsart-alias  
/VERTRAGSPARTNER_KTONR"/>
```

```
<variable ref="n_KontoklasseID_Kokl">
```

```
<get-id-from container="KONTOKLASSE">
```

```
<variable ref="v_Bezeichnung" multi-mapping="buchungsart-alias  
/VERTRAGSPARTNER_KTOKLASSE"/>
```

```
</get-id-from>
```

```
</variable>
```

```
</get-id-from>
```

```
</variable>
```

```
...
```

```
</import-container>
```

```
</import-container>
```

```
</import-schema>
```

```
<multi-mapping name="buchungsart-alias" href="Multi-Mappingbuchungsart-  
mm.xml">
```

```
<source ref="BA_NR_ALT" field="BA_NR"/>
```

```
<source ref="ANL_NR_BEL_ALT" field="ANL_NR_BEL"/>
```

```
<source ref="ANL_NR_HER_ALT" field="ANL_NR_HER"/>
```

```
<source ref="DUPLIZIER_ALT" field="DUPLIZIER"/>
```

```
<source ref="STO_GRD_ALT" field="STO_GRD"/>
```

```
</multi-mapping>
```

```
...
```

```
</interface>
```


10 Export

Die Neuanlage einer Exportroutine „BSP_EXPORT“ mit der Bezeichnung „BSP“ führt zu Anlage der drei bekannten Templates. Die grundsätzliche Vorgehensweise ist im [Kapitel 3](#) bereits beschrieben.

10.1. Einbinden Multi - Mappings im Export

Auch im Export ist die Umschlüsselung anhand der Verwendung von Multi - Mappings möglich. Dazu ist das Multi-Mapping in der BSP-interface.xml einzubinden und zu referenzieren.

In der Standardversion wird „Retourengrund“ wie folgt ausgegeben:

```
<field ref="RetourenID_Retg">
  <field ref="Kennnummer" variable="n_Kennnummer"
  <follow-forward container="RETOUREN" via="n_RetourenID_Retg"/>
</field>
```

Zum besseren Verständnis sei dies erläutert:

- Löse RetourenID_Retg auf
- Fülle Feld „Kennnummer“ mit dem Wert der Variable „n_Kennnummer“
- „n_Kennnummer“ muss über die Auflösung der „n_RetourenID_Retg“ in der Stammdatentabelle „TVW_VWRETOUREN“ ausgelesen werden

Variable „n_Kennnummer“ dient dazu, das Multi-Mapping zu ersetzen und in der Referenzierung des Multi-Mappings am Ende des interface.xml einzusetzen. „Kennnummer“ wird zum besseren Verständnis umbenannt in „ALTES_RETOURKZ“.

```
<field ref="RetourenID_Retg">
  <field ref="ALTES_RETOURKZ" multi-mapping="retouralias/RETOUR_ALT"/>
  <follow-forward container="RETOUREN" via="n_RetourenID_Retg"/>
</field>
```

Entsprechend der bereits dargestellten Multi-Mapping-Funktionalität ist im Interface die entsprechende Einbindung vorzunehmen (am Ende anhängen). Allerdings wird im Gegensatz zum Import auf <target> referenziert. Die Variable „n_Kennnummer“ aus dem ursprünglichen RetourenID_Retg wird im Definitionsteil des Multi-Mappings gesetzt.

```
<multi-mapping name="retouralias" href="c:\wme.fact\schnittstelle\multimap\retour.xml">
  <target ref=" VWRETOUREN KENNUMMER" field="n_Kennnummer"/>
</multi-mapping>
```

Damit wird folgende Schleife beim Export abgearbeitet:

- Löse RetourenID_Retg auf
- Fülle Feld „ALTES_RETOURKZ“ mit dem Wert der (Source-)Variablen „RETOUR_ALT“ aus dem Multi - Mapping „retouralias“
- Im „retouralias“ ist hinterlegt, das der Wert der Variablen „n_Kennnummer“ (aus RetourenID_RETG) der (Target-)Variablen „VWRETOUREN KENNUMMER“ entsprechen soll
- „n_Kennnummer“ wird über „n_RetourenID_Retg“ in der Stammdatentabelle „TVW_VWRETOUREN“ aufgelöst, der entsprechende Eintrag „VWRETOUREN KENNUMMER“ ermittelt und der Wert der Variablen „RETOUR_ALT“ ausgelesen
- RETOUR_ALT wird wieder übergeben an das Feld „ALTES_RETOURKZ“

11 Anhang

11.1. Abbildungsverzeichnis

Abbildung 1-1: Schema Importvorgang 9

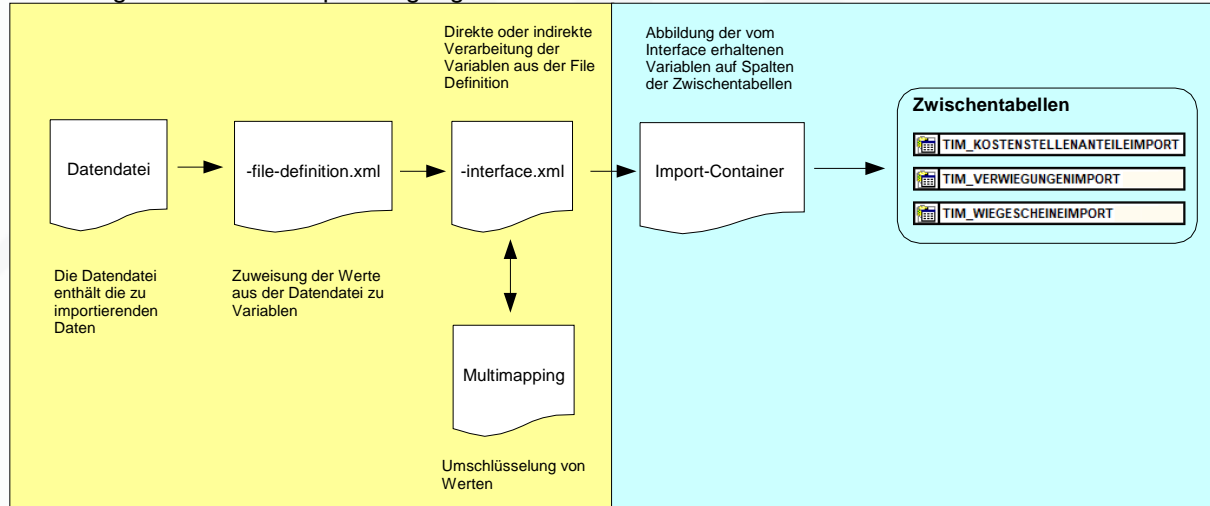


Abbildung 2-1: Schema XML - Dateien, Zwischentabellen 10

Abbildung 2-2: Dialog Waage Schnittstelle Import (Erstellen) 11

Abbildung 2-3: WsBA31Beispiel-import.xml 15

Abbildung 2-4: BA31-file-definition.xml (Ausschnitt) 16

Abbildung 2-5: BA31-interface.xml (Ausschnitt) 17

Abbildung 2-6: Multi-Mapping retour-mm.xml 19

Abbildung 2-7: Dialog Waage Schnittstelle Import (Importieren) 21

Abbildung 3-1: Dialog Waage Schnittstelle Export 23

Abbildung 3-2: -file-definition.xml (Export) 24

Abbildung 4-1: Dialog Waage Schnittstelle Pflege 25

Abbildung 4-2: Dialog Waage Schnittstelle Löschen 26

Abbildung 4-3: Dialog Waage Schnittstelle Übernahme 28

Abbildung 5-1: Multi - Mapping buchungsart-mm.xml (Ausschnitt) 34

Abbildung 5-2: Darstellung Abhängigkeiten Artikel 36


Abbildung 5-3: BA31-interface.xml (Ausschnitt) 39

Abbildung 9-1: Multi-Mapping buchungsart-mm.xml 63

11.2. Variablen

Kernstück einer Routine ist die Schnittstellendefinition (-interface.xml). Dort erfolgt die Zuordnung von Quell- zu Zielvariablen. In den folgenden Tabellen wird dargestellt, welche Daten überhaupt eingelesen werden können bzw. müssen.

Hinweis zu WME - Name und WME – Pos.

 Die in den folgenden Tabellen angegebenen Spalten WME - Name und WME – Pos. dienen der Orientierung bei der Zuordnung der alten WME - Nomenklatur zur der neuen Nomenklatur von w^me.fact. In einigen Fällen sind die in den Spalten enthaltenen Felder nicht kompatibel, sondern sind nur als Anhaltspunkt zu verstehen.

Die Namens- und Positionsangaben entsprechen den im Dokument „Schnittstellenbeschreibung zu WME-Neu“ (Bestell-Nr. 205.D4) enthaltenen Schnittstellen „B.1 Input-/Output-Daten“ (BA31/BA32).

Felder (Zeilen) ohne Inhalt in den Spalten WME - Name und WME – Pos. sind mit w^me.fact neu eingeführt worden und haben demzufolge keine Entsprechung in der alten Nomenklatur.

-interface.xml import-container TIM_VERWIEGUNG	Anzugebender Wert	Beschreibung nach neuer Schnittstelle	WME-Name BA31/32	WME-Pos BA31/32
<variable ref="BUCHUNGSART">	n_Kennnummer	Kennnummer der Buchungsart, möglich sind Buchungsarten der Eingangs- und Ausgangsverwiegung	ZUSTAENDIG	178
<variable ref="VERTRAGSPARTNER">	v_Kontonummer <u>und</u> n_Kennnummer (Kontoklasse)	Kontonummer des beteiligten Personenkontos und Kennnummer der Kontoklasse dieses Personenkontos	ANL_NR_HER/ EMPFAENGER/ ANLCO_EMPF	41-46 90-95 179-184
<variable ref="AUFTRAG">	v_Auftragsnummer <u>und</u> n_Kennnummer (Auftragsart) v_Kontonummer <u>und</u> n_Kennnummer (Kontoklasse) des Garantiegebers	Wenn Auftragsnummer der DSD zu einem Vorgang importiert werden soll, so muss Auftragsnummer und Kennnummer der Auftragsart angegeben werden Optional: Kontonummer und Kontoklasse des Garantiegebers DSD	DISPO_NR	109-116
<variable ref="ARTIKEL">	v_Artikelnummer	Artikelnummer wme.fact	FRAKT_NR	31-34
<variable ref="ARTIKELLAGERORT">	v_Artikelnummer <u>und</u> v_Kurzname (Lagerort)	Artikellagerort muss immer zum Artikel angegeben werden; wird angegeben durch die Kombination von Artikel und Lagerort		
<variable ref="RETOURE">	n_Kennnummer	Kennnummer des Retourengrundes	RETOUR_KZ	187
<variable ref="USERWERT">	v_Bezeichnung (Dialog) <u>und</u> v_Userwert	Bezeichnung des Dialogs (siehe 11.4.20) und ein dem Dialog zugeordneter Userwert	???	???
<variable ref="GARANTIEGEBER">	v_Kontonummer <u>und</u>	Kontonummer und Kennnummer (Kontoklasse)	ANLCO_EMPF	179-184

	n_Kennnummer (Kontoklasse) des Garantiegebers	des für die Menge zuständigen Garantiegeber (Bitte nicht verwechseln mit Garantiegeber bei Aufträgen DSD)	(nicht Entsorger / Deponie / MVA)	
<variable ref="AUFTRAGSNUMMER">	(Auftragsnummer)	Freie oder interne Auftragsnummer (nicht Auftragsnummer DSD)		
<variable ref="BELEGDATUM">	(Datum)	Datum des Beleges (Verwiegungsmaster), gibt das tatsächliche Datum des Wareneingangs / -ausgangs der Mengen dieser Verwiegung an, i.d.R. identisch mit Wiegescheindatum	WIEGE_DAT (kann)	68-75
<variable ref="EINGANG">	T oder F	Wareneingang (=T) oder Warenausgang (=F)	BA_NR	01-02

-interface.xml	Anzugebender Wert	Beschreibung nach neuer Schnittstelle	WME-Name BA31/32	WME-Pos BA31/32
import-container TIM_WIEGESCHEIN				
<variable ref="SENDERWS">	T oder F	Senderwiegeschein (=T) oder Empfängerwiegeschein (=F)		
<variable ref="WIEGESCHEIN_NR">	(Wiegescheinnummer)	Wiegescheinnummer (wird durch Bearbeitungsvorschriften im Template um Sonderzeichen bereinigt)	WIEGE_NR	48-67
<variable ref="BEFOERDERUNGSART">	n_Kennnummer	Kennnummer der Beförderungsart	BEFOERD	161-162
<variable ref="DATUM">	(Datum)	Datum der Wiegescheins [tt.mm.yyyy]; darf max. 10 Tage vom Belegdatum des Masters abweichen	WIEGE_DAT	68-75
<variable ref="BRUTTO">	(Bruttogewicht Wiegeschein)	Bruttogewicht in [to], abweichend zur alten Schnittstelle müssen Nachkommastellen mit <i>Komma</i> angegeben werden	BRUTTO	117-125
<variable ref="TARA">	(Taragewicht Wiegeschein)	Taragewicht in [to], abweichend zur alten Schnittstelle müssen Nachkommastellen mit <i>Komma</i> angegeben werden	TARA	130-138
<variable ref="VOLUMEN">	(Volumen)	Volumen in [cbm]; muss zum Gebinde des Artikels passen	VOLUMEN	81-89
<variable ref="ANZAHL">	(Anzahl)	Anzahl in [Stück]; muss zum Gebinde des Artikels passen	BALLEN_ANZ	76-80
<variable ref="BRUTTOZEIT">	(Uhrzeit der Bruttoverwiegung)	Brutto Uhrzeit [hh:mm]	BRUTTOZEIT	126-129
<variable ref="TARAZEIT">	(Uhrzeit der Taraverwiegung)	Tara Uhrzeit [hh:mm]	TARA_ZEIT	139-142
<variable ref="NETTOKORREKTUR">	(Nettokorrektur)	Nettokorrekturgewicht in [to]		
<variable ref="VWSTATUS">	n_Kennnummer	Kennnummer des Verwiegestatus		

<variable ref="LAND">	v_Laenderkennzeichen	Länderkennzeichen des Transportfahrzeugs		
<variable ref="KFZ">	v_Kuerzel	Kurzzeichen des Kreises des Transportfahrzeugs (Bitte Plausibilitäten in Abhängigkeit der Beförderungsart beachten)	KFZ	163-177
<variable ref="KENNZEICHENFREI">	(Rest des Kennzeichens)	Hinterer Teil des KFZ-Kennzeichens; (wird durch Bearbeitungsvorschriften im Template um Sonderzeichen bereinigt)	KFZ	163-177

-interface.xml import-container KOSTENSTELLENANTEIL	Anzugebender Wert	Beschreibung nach neuer Schnittstelle	WME-Name BA31/32	WME-Pos BA31/32
<variable ref="ANTEIL">	(Anteil)	Anteil in Prozent, Nachkommastellen müssen durch <i>Komma</i> getrennt sein	ANTEIL_VG	152-160
<variable ref="KOSTENSTELLENFALL">	n_Kennummer	Kennummer des Kostenstellenfall		
<variable ref="KOSTENSTELLE1">	V_Kuerzel (Kostenstelle) d_GueltigAb (Kostenstelle) n_Kennummer (Kostenstellenart)	Angabe der 1. Kostenstelle; in Abhängigkeit vom Kostenstellenfall sind nur Kostenstellen einer bestimmten Kostenstellenart zulässig; die Kostenstelle ist anzugeben mit Kuerzel, GültiAb-Datum und Kennummer der Kostenstellenart	VG_NR	03-07
<variable ref="KOSTENSTELLE2">	s.o. analog	s.o. analog		
<variable ref="KOSTENSTELLE3">	s.o. analog	s.o. analog		

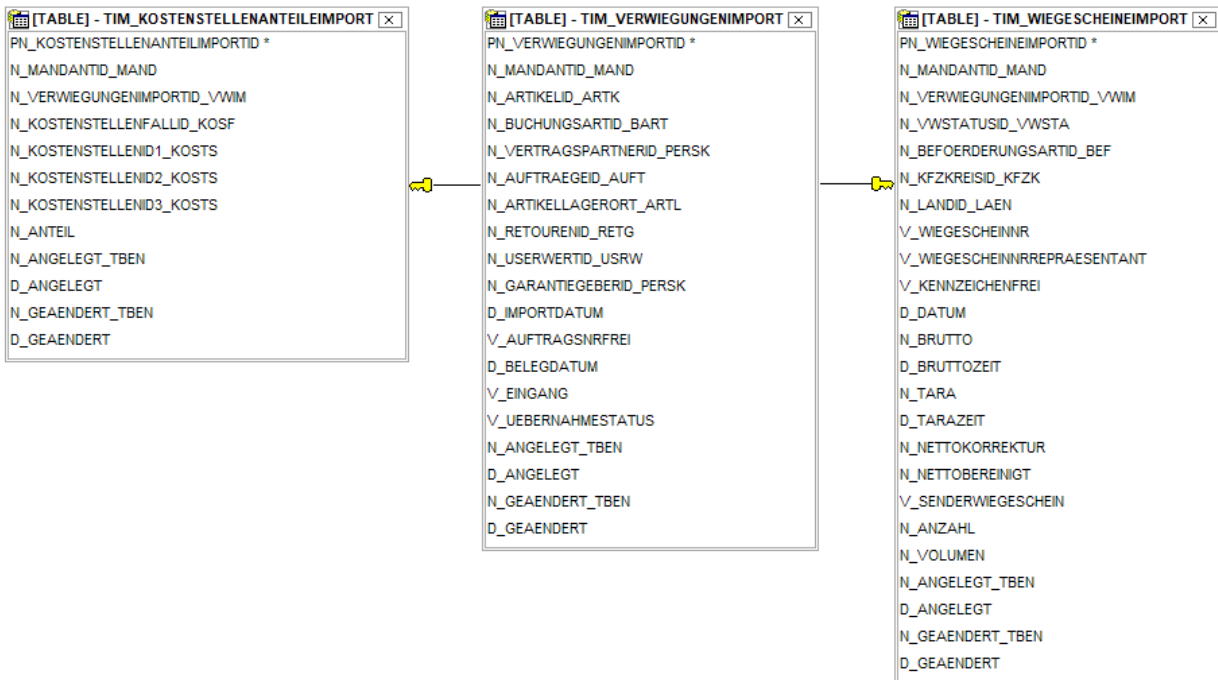
Legende:

n_Variable	erwartet wird ein numerischer Wert, der auf einen Eintrag in den Systemtabellen wme.fact referenziert
v_Variable	erwartet wird ein alphanumerischer Wert, der auf einen Eintrag in den Systemtabellen wme.fact referenziert
d_Variable	erwartet wird ein Datumswert, der auf einen Eintrag in den Systemtabellen wme.fact referenziert
(Variable):	freie Werte ohne Referenz auf die Systemtabellen

Jeder der obigen Tabellen entspricht einem „import-container“ - Element in der -interface.xml.

Die Definition aller Variablen ist der Datei <wme.fact>\Config\Transfer\import-container.xml zu entnehmen. Hier werden alle Variablen definiert, auf die in -interface.xml referenziert werden kann.

11.3. Zwischentabellen



Die Zwischentabellen (TIM_...) sind analog der Tabellenstruktur der Echttabellen (TVW_...) aufgebaut.

Mastertabelle ist die TIM_VERWIEGUNGENIMPORT, die dem Master Eingangs- bzw. Ausgangsverwiegung in ^{wm}e.fact entspricht.

In der TIM_WIEGESCHEINEIMPORT können jedem Verwiegungsmaster n Wiegescheindaten zugeordnet werden (siehe im Dialog die Details „Senderwiegescheine“ und „Empfängerwiegescheine“).

In der TIM_KOSTENSTELLENANTEILEIMPORT werden die unterschiedlichen Kostenstelleninformationen zum Master gespeichert.

Die Präfix der Variablen kennzeichnet das Format:

- N numerischer Wert
- D Datum
- V Varchar
- B Boolean
- PN unique Key

Ist an die Variable eine Suffix im Format ID_xxxx angehängt, so enthält die Variable die Referenz auf eine Fremdtabelle und ist beim Import von Daten entsprechend über den Unique Key der Fremdtabelle aufzulösen.

11.4. Listen

Nachfolgend beschrieben sind die in den Variablen der -interface.xml verwendeten Fremdschlüssel-tabellen. Diese können auch mit Datenbanktools und direktem, lesenden Zugriff auf die Datenbank (Name: <wme.fact>\database\mebus.gdb, Benutzer: report, Passwort: report, Datenbankversion: Firebird 1.5) selbst ermittelt werden.

Die Daten entsprechen der Stammdatenversion E-U-121-20050316-000076, bzw. E-I-121-20050316-000076.

11.4.1. Liste Beförderungsarten (TVW_BEFOERDERUNGSARTEN)

- interface.xml: TIM_WIEGESCHEIN.BEFOERDERUNGSART

Kennnummer	Beschreibung	Min. Tara [to]	Max-Tara [to]	Min. Netto [to]	Max. Netto [to]	WME: BEFOERD
10	Pressmüllfahrzeug	5,000	30,000	0,001	20,000	½
11	Containerzug	5,000	40,000	0,001	30,000	½
12	Sattelauflieger	5,000	40,000	0,001	35,000	½
13	Walking Floor Fahrzeug	5,000	30,000	0,001	30,000	½
14	Multilift Fahrzeug	5,000	30,000	0,001	20,000	½
15	Absetz-/ Abrollkipper	5,000	20,000	0,001	20,000	½
16	Hänger	2,000	10,000	0,001	20,000	½
17	Pritschenfahrzeug	0,500	8,000	0,001	10,000	½
18	LKW, Sonstiges	0,500	50,000	0,001	30,000	½
20	Transporter	0,500	8,000	0,001	10,000	½
30	Schiff	0,001	10.000,000	0,001	1.000,000	05/06
40	Zug	0,001	1.000,000	0,001	100,000	¾
50	Stapler	0,200	15,000	0,001	2,000	
60	Intern	0,001	10,000	0,001	500,000	20/99

11.4.2. Liste Retourengründe (TYF_VWRETOUREN)

- interface.xml: TIM_VERWIEGUNG.RETOURE

Kennnummer	Beschreibung	WME: RETOUR_KZ
1	[keine Retoure]	F
2	Nicht spezifikationsgerecht	T

11.4.3. Liste Status Wiegeschein (TYF_VWSTATUS)

- interface.xml: TIM_WIEGESCHEIN.VWSTATUS

Kennnummer	Beschreibung
1	EB (Elektronischer Beleg)
2	EB mit hinterlegtem Tara
3	EB mit man. Gewichtsänd.
4	GB (Gestanzter Beleg)

11.4.4. Liste Buchungsarten (TYF_BUCHUNGSARTEN)

- interface.xml: TIM_VERWIEGUNG.BUCHUNGSART

Kennnummer	Beschreibung	Bereich	WME: ZUSTAENDIG
100	EV aus Haushaltssammlung	Wareneingang	
101	EV aus Gewerbesammlung	Wareneingang	
102	EV von Umschlaganlage	Wareneingang	
103	EV von Sortieranlage	Wareneingang	
104	EV von Lager	Wareneingang	
105	EV von Händler	Wareneingang	
106	EV von Aufbereitungsanlage	Wareneingang	
107	EV von Verwertungsanlage	Wareneingang	
108	AV an Umschlaganlage	Warenausgang	1
109	AV an Sortieranlage	Warenausgang	1
110	AV an Lager	Warenausgang	2
111	AV an Händler	Warenausgang	2
112	AV an Aufbereitungsanlage	Warenausgang	2
113	AV an Verwertungsanlage	Warenausgang	2
114	AV an MVA / MBA	Warenausgang	3
115	AV an Deponie	Warenausgang	3
116	AV an Sonst. Restanlagen	Warenausgang	3
117	EV aus Pfandsammlung	Wareneingang	

11.4.5. Liste Kontoklassen (TYF_KONTOKLASSEN)

Kennnummer	Bezeichnung	Kontogruppe. Kennnummer	Kontogruppe.Bezeichnung
2	Anlieferer		
3	Garantiegeber		
4	Pfandsammelstellen		
21	Umschlaganlage	100	Entsorger Anlagen
22	Sortieranlage	100	Entsorger Anlagen
23	Lager	200	Verwerter Anlagen
24	Händler	200	Verwerter Anlagen
25	Aufbereitungsanlage	200	Verwerter Anlagen
26	Verwertungsanlage	200	Verwerter Anlagen
27	MVA / MBA		
28	Deponie		
29	Sonstige Restanlagen		

11.4.6. Liste Kostenstellenfälle (TYF_KOSTENSTELLENFAELLE)

- interface.xml: KOSTENSTELLENANTEIL.KOSTENSTELLENFALL

Kennnummer	Beschreibung	Bezeichnung Detail in Dialogen wme.fact
1	LV / Kommune	Kommune
2	LV / Systembetreiber	Kostenträger (hier eingeschränkt auf Systembetreiber)
6	LV / Rücknahmesystem	Kostenträger (hier eingeschränkt auf „61696“)

11.4.7. Liste Kostenstellenarten (TYF_KOSTENSTELLENARTEN)

- interface.xml: KOSTENSTELLENANTEIL.KOSTENSTELLENART

Kennnummer	Bezeichnung
2	Garantiegeber
3	Leistungsvertrag
4	Kommune
5	Sammelsystem
6	Systembetreiber
7	ohne
8	Rücknahmesystem

11.4.8. Liste Personenkonten (TST_PERSONENKONTEN)

- interface.xml: TIM_VERWIEGUNG.VERTRAGSPARTNER
- interface.xml: TIM_VERWIEGUNG.GARANTIEGEBER

Kontoklasse (Referenz auf TYF_KONTOKLASSEN)	Kontonummer
Sortieranlage	2402
...	...

11.4.9. Liste Kostenstellen (TYF_KOSTENSTELLEN)

- interface.xml: KOSTENSTELLENANTEIL.KOSTENSTELLE1
- interface.xml: KOSTENSTELLENANTEIL.KOSTENSTELLE2
- interface.xml: KOSTENSTELLENANTEIL.KOSTENSTELLE3

Kostenstellenart (Referenz)	Kuerzel	Bezeichnung
Leistungsvertrag	NW005-2004L0-101	Kostenstelle der Kostenstellenart Leistungsvertrag
...

11.4.10. Liste Artikel (TST_ARTIKEL)

- interface.xml: TIM_VERWIEGUNG.ARTIKEL

Artikelgruppe (Referenz Feinfraktion)	Artikelnummer (frei erfassbar)	Artikelbezeichnung
0600	0600	LVP (wie gesammelt)
...

11.4.11. Liste Lagerort (TST_LAGERORTE)

Lagerort (Kurzname) (frei erfassbar)	Bezeichnung
Halle1	Halle
...	...

11.4.12. Liste Artikellagerort (TST_ARTIKELLAGERORTE)

- interface.xml: TIM_VERWIEGUNG.ARTIKELLAGERORT)

Lagerort (Referenz auf Kurzname)	Artikel (Referenz auf Artikel)
L100	0100/1
...	...

11.4.13. Liste Aufträge (TTR_AUFTRAEGE)

- interface.xml: TIM_VERWIEGUNG.AUFTRAG

Zu unterscheiden ist zwischen Auftragsnummern DSD (Variable „AUFTRAG“) und der freien Auftragsnummer (Variable „AUFTRAGSNUMMER“). Für die Zuordnung eines DSD - Auftrages in der Waage-schnittstelle sind anzugeben:

- Auftragsnummer = z.B. 101010 (immer 6 – stellig)
- Auftragsart = derzeit nur DSD
- Garantiegeber des Auftrags = Personenkonto des Garantiegebers; derzeit Kontonummer 3 (DSD) und Kontoklasse 3 (Garantiegeber)

Auftragsart (Referenz auf TYF_AUFTRAGSARTEN)	Auftragsnummer
DSD	101010
...	...

11.4.14. Liste Kreis (TYF_KFZKREISE)

- interface.xml: TIM_WIEGESCHEIN.KREIS

Kuerzel	Kreis
-	Kein (z.B. bei ausländischen KFZ)
A	Augsburg (Stadt und Land)
...	...

11.4.15. Liste Land (TYF_LAENDER)

- interface.xml: TIM_WIEGESCHEIN.LAND

Länderkennzeichen	Land
A	Österreich
...	...

11.4.16. Liste Garantiegeber über Personenkonto (TST_PERSONENKONTEN)

- interface.xml: TIM_ VERWIEGUNG.GARANTIEGEBER

Kontonummer	Kontoklasse	Kurzname
1	3	DAVR
2	3	GGA
3	3	DSD
5	3	Interseroh
6	3	Vfw
7	3	ReCarton
10	3	TSR
11	3	Recostra
12	3	Rasselstein Hoesch
13	3	GEBR
14	3	DGW
15	3	Sita
16	3	ZENTEK
2146	3	Merkator
2179	3	VGS
2940	3	THYSSEN ALFA
3250	3	RHM
48117	3	[kein Garantiegeber]
50363	3	Interseroh Hansa
50368	3	Remondis
50397	3	DWR
53864	3	Eigennachweis
63735	3	[Fremdmengen]
66860	3	[Naturmaterialien]
78092	3	REMONDIS Trade

11.4.17. Liste Garantiegeber über Kostenstelle (TST_KOSTENSTELLEN)

- interface.xml: TIM_ VERWIEGUNG.GARANTIEGEBER

Kürzel	Kostenstellenart	Bezeichnung
209	2	[Fremdmengen] (Fremdmengen)
30	2	[kein Garantiegeber] (Sonstige)
1	2	DAVR (Aluminium)
17	2	DGW (Aluminium)
19	2	DGW (Weißblech)
3	2	DSD (Kunststoff)
109	2	DWR (Aluminium)
102	2	DWR (Weißblech)
189	2	Eigennachweis (Aluminium)
190	2	Eigennachweis (Glas)
130	2	Eigennachweis (Kunststoff)
193	2	Eigennachweis (Papier)
191	2	Eigennachweis (Sonst. Verbunde)
192	2	Eigennachweis (Weißblech)
16	2	GEBR (Weißblech)
2	2	GGA (Glas)
12	2	Interseroh (Aluminium)
91	2	Interseroh (Glas)
13	2	Interseroh (Sonst. Verbunde)
11	2	Interseroh (Weißblech)
89	2	Interseroh Hansa (Weißblech)

Kürzel	Kostenstellenart	Bezeichnung
96	2	Merkator (Aluminium)
97	2	Merkator (Weißblech)
15	2	Rasselstein Hoesch (Weißblech)
7	2	ReCarton (Getränkekartons)
28	2	Recostra (Aluminium)
94	2	Remondis (Aluminium)
95	2	Remondis (Weißblech)
101	2	RHM (Weißblech)
20	2	Sita (Aluminium)
22	2	Sita (Weißblech)
170	2	THYSSEN ALFA (Aluminium)
100	2	THYSSEN ALFA (Weißblech)
10	2	TSR (Weißblech)
23	2	Vfw (Aluminium)
92	2	Vfw (Glas)
24	2	Vfw (Weißblech)
98	2	VGS (Aluminium)
99	2	VGS (Weißblech)
25	2	ZENTEK (Aluminium)
93	2	ZENTEK (Glas)
27	2	ZENTEK (Weißblech)

11.4.18. Abhängigkeit Retourkennzeichen

Die Validierung des Retourkennzeichens wird gesteuert über TYF_BARTZUKONTOGRUPPEN.B_RETOURKZ. Bei Eintrag „T“ ist ein Retourkennzeichen ungleich „keine Retoure“ anzugeben.

1.1.21 Eingangsverwiegunen

TYF_BUCHUNG SARTEN. N_KENNUMMER	TYF_BUCHUNGSARTEN. V_BUCHUNGSART	TYF_BARTZUKONTO GRUPPEN. B_RETOURKZ	TYF_KONTOGRUPPEN. V_BEZEICHNUNG
100	EV aus Haushaltssammlung	F	Entsorger Anlagen
100	EV aus Haushaltssammlung	F	Verwerter Anlagen
101	EV aus Gewerbesammlung	F	Entsorger Anlagen
101	EV aus Gewerbesammlung	F	Verwerter Anlagen
102	EV von Umschlaganlage	F	Entsorger Anlagen
102	EV von Umschlaganlage	F	Verwerter Anlagen
103	EV von Sortieranlage	F	Entsorger Anlagen
103	EV von Sortieranlage	F	Verwerter Anlagen
104	EV von Lager	T	Entsorger Anlagen
104	EV von Lager	F	Verwerter Anlagen
105	EV von Händler	T	Entsorger Anlagen
105	EV von Händler	F	Verwerter Anlagen
106	EV von Aufbereitungsanlage	T	Entsorger Anlagen
106	EV von Aufbereitungsanlage	F	Verwerter Anlagen
107	EV von Verwertungsanlage	T	Entsorger Anlagen
107	EV von Verwertungsanlage	F	Verwerter Anlagen

1.1.22 Ausgangsverwiegunen

TYF_BUCHUNG SARTEN. N_KENNUMMER	TYF_BUCHUNGSARTEN. V_BUCHUNGSART	TYF_BARTZUKONTO GRUPPEN. B_RETOURKZ	TYF_KONTOGRUPPEN. V_BEZEICHNUNG
108	AV an Umschlaganlage	F	Entsorger Anlagen
108	AV an Umschlaganlage	T	Verwerter Anlagen
109	AV an Sortieranlage	F	Entsorger Anlagen
109	AV an Sortieranlage	T	Verwerter Anlagen
110	AV an Lager	F	Entsorger Anlagen
110	AV an Lager	F	Verwerter Anlagen
111	AV an Händler	F	Entsorger Anlagen
111	AV an Händler	F	Verwerter Anlagen
112	AV an Aufbereitungsanlage	F	Entsorger Anlagen
112	AV an Aufbereitungsanlage	F	Verwerter Anlagen
113	AV an Verwertungsanlage	F	Entsorger Anlagen
113	AV an Verwertungsanlage	F	Verwerter Anlagen
114	AV an MVA / MBA	F	Entsorger Anlagen
114	AV an MVA / MBA	F	Verwerter Anlagen
115	AV an Deponie	F	Entsorger Anlagen
115	AV an Deponie	F	Verwerter Anlagen
116	AV an Sonst. Resteanlagen	F	Entsorger Anlagen
116	AV an Sonst. Resteanlagen	F	Verwerter Anlagen

11.4.19. Liste Artikelgruppen (TYF_ARTIKELGRUPPEN)

Feinfraktion	Bezeichnung
0100	Mischglas
0101	Weißglas
0102	Grünglas
0103	Braunglas
0104	Buntglas
0105	Glas aus LVP
0106	3-farb erfaßt; gem. verwogen
0107	weiß/grün erfaßt;gem. verwogen
0108	grün/braun erfaßt;gem.verwogen
0109	weiß/braun erfaßt;gem.verwogen
0200	PPK aus Sammelmenge ÖRE
0201	1.02/B12 sort. Gemischpapier
0202	1.04/B19 Kaufhausaltpapier
0203	4.03/W52 gebr.Kraftwellpappe
0204	1.01/B10 unsort. Gemischpapier
0205	1.03/H12 Graukarton
0206	5.01/A00 gemischtes Altpapier
0207	PPK aus LVP
0250	PPK aus Gewerbemenge § 6(3)
0300	Kunststoffe
0310-0	Folien
0320-0	Flaschen
0321-0	PO Flaschen
0322-0	Hohlkörper > 5 l
0324-0	PP (Polypropylen)
0325-0	PET Flaschen transparent
0325-1	PET Flaschen klar
0325-2	PET Flaschen bunt
0328-0	PET Gemisch
0329-0	PE (Polyethylen)
0330-0	Becher
0331-0	PS (Polystyrol)
0340-0	EPS
0340-1	EPS verpreßt
0340-2	EPS Briketts
0340-4	EPS Chips
0350-0	Mischkunststoffe
0350-4	MK - Polyolefinreich
0350-5	MK – Polyolefinarm
0350-6	MK – weich
0350-7	MK – hart
0350-90	MK – Chlor abgereichert
0350-91	MK - stark Chlor abgereichert
0351-0	Formstabile Kunststoffe
0355-0	MK außerhalb DSD Spezifikation
0360	stoffgleiche Kunststoffe
0400	Metall
0410	Weißblech
0420	Aluminium
0421	Alu-Output (VE);Reinheit >=95%
0422	Alu-Output (VE);Reinheit < 95%
0423	Alu-Verb.-Reject (energ.Verw.)
0424	Gemisch Alu-Verb.-Reject
0425	Aluminium aus anderen Samml.

Feinfraktion	Bezeichnung
0430	stoffgleiches Metall aus LVP
0435	stoffgleiches Alu aus LVP
0440	Elektronikschrott
0510	5.03 Getränkeverbunde aus LVP
0520	Sonstige Verbunde aus LVP
0600	LVP (wie gesammelt)
0610	Flach
0615	Rund
0620	Kunststoffe und Verbunde
0625	Glas und Metalle
0630	LVP und PPK
0635	Biotonne
0640	Glas, Metall, Getränkeverbunde
0700	Naturmaterialien
0710	Kork
0720	Steingut
0730	Textilien
0740	Holz
0830	Sortierreste aus LVP
0840	Sortierreste aus Glas u.Metall
0841	Sortierreste Glas/Metall/FKN
0845	Sortierreste aus Metall
0850	Sortierreste aus Flach u. Rund
0870	Sort.reste aus Kunstst.+ Verb.
0875	Sortierreste aus Kunststoffe
0880	Sortierreste aus LVP/PPK Misch
0910	Aufbereitungsreste Glas
0920	Aufbereitungsreste Kunststoff
0930	Aufbereitungsreste Weißblech
0940	Aufbereitungsreste Aluminium

11.4.20. Liste Dialoge (TYT_DIALOGE)

- interface.xml: TIM_VERWIEGUNG.USERWERT

Bezeichnung	Dialogname
DLG_Wareneingang	Eingangsverwiegung
DLG_Warenausgang	Ausgangsverwiegung

11.4.21. Liste Buchungsarten - Kostenstellenfälle

- interface.xml: TIM_KOSTENSTELLENANTEIL.KOSTENSTELLENFALL

Kontogruppe Entsorgeranlagen

Buchungsart	Bezeichnung	KStFall	Bezeichnung	Pflicht
100	EV aus Haushaltssammlung	1	LV / Kommune	F
100	EV aus Haushaltssammlung	6	LV / Rücknahmesystem	T
101	EV aus Gewerbesammlung	1	LV / Kommune	F
101	EV aus Gewerbesammlung	6	LV / Rücknahmesystem	T
102	EV von Umschlaganlage	6	LV / Rücknahmesystem	T
103	EV von Sortieranlage	6	LV / Rücknahmesystem	T
104	EV von Lager	2	LV / Systembetreiber	T
105	EV von Händler	2	LV / Systembetreiber	T
106	EV von Aufbereitungsanlage	2	LV / Systembetreiber	T
107	EV von Verwertungsanlage	2	LV / Systembetreiber	T
108	AV an Umschlaganlage	6	LV / Rücknahmesystem	T
109	AV an Sortieranlage	6	LV / Rücknahmesystem	T
110	AV an Lager	2	LV / Systembetreiber	T
111	AV an Händler	2	LV / Systembetreiber	T
112	AV an Aufbereitungsanlage	2	LV / Systembetreiber	T
113	AV an Verwertungsanlage	2	LV / Systembetreiber	T
114	AV an MVA / MBA	6	LV / Rücknahmesystem	T
115	AV an Deponie	6	LV / Rücknahmesystem	T
116	AV an Sonst. Resteanlagen	6	LV / Rücknahmesystem	T

Kontogruppe Verwerteranlagen

Buchungsart	Bezeichnung	KStFall	Bezeichnung	Pflicht
100	EV aus Haushaltssammlung	1	LV / Kommune	F
100	EV aus Haushaltssammlung	2	LV / Systembetreiber	T
101	EV aus Gewerbesammlung	1	LV / Kommune	F
101	EV aus Gewerbesammlung	2	LV / Systembetreiber	T
102	EV von Umschlaganlage	2	LV / Systembetreiber	T
103	EV von Sortieranlage	2	LV / Systembetreiber	T
104	EV von Lager	2	LV / Systembetreiber	T
105	EV von Händler	2	LV / Systembetreiber	T
106	EV von Aufbereitungsanlage	2	LV / Systembetreiber	T
107	EV von Verwertungsanlage	2	LV / Systembetreiber	T
108	AV an Umschlaganlage	2	LV / Systembetreiber	T
109	AV an Sortieranlage	2	LV / Systembetreiber	T
110	AV an Lager	2	LV / Systembetreiber	T
111	AV an Händler	2	LV / Systembetreiber	T
112	AV an Aufbereitungsanlage	2	LV / Systembetreiber	T
113	AV an Verwertungsanlage	2	LV / Systembetreiber	T
114	AV an MVA / MBA	2	LV / Systembetreiber	T
115	AV an Deponie	2	LV / Systembetreiber	T
116	AV an Sonst. Resteanlagen	2	LV / Systembetreiber	T

11.4.22. Liste Kostenstelle1

- interface.xml: TIM_KOSTENSTELLENANTEIL.KOSTENSTELLE1

Für Kostenstellenfall	Kürzel	Gültig ab	Kostenstellenart
Für alle	Kürzel des Leistungsvertrages, z.B. AB010-2004CD-101	Dem jeweiligen Vertrag zu entnehmen	3

11.4.23. Liste Kostenstelle2

- interface.xml: TIM_KOSTENSTELLENANTEIL.KOSTENSTELLE2

Für Kostenstellenfall	Kürzel	Gültig ab	Kostenstellenart
LV / Kommune	Amtlicher Gemeindeschlüssel der Kommune, z.B. 01056041	Der jeweiligen Kommune zu entnehmen	4
LV / Rücknahmesystem	Immer „61696“	01.01.2004	8
LV / Systembetreiber	Kürzel des Systembetreibers, siehe Anhang 11.4.26	Siehe Anhang 11.4.26	6

11.4.24. Liste Kostenstelle3

- interface.xml: TIM_KOSTENSTELLENANTEIL.KOSTENSTELLE3

Für Kostenstellenfall	Kürzel	Gültig ab	Kostenstellenart
LV / Kommune	Kürzel des Sammelsystems, siehe Anhang 11.4.25	siehe Anhang 11.4.25	5
LV / Rücknahmesystem	Immer „ohne“	01.01.1990	7
LV / Systembetreiber	Immer „ohne“	01.01.1990	7

11.4.25. Liste Sammelsysteme

Kürzel	Bezeichnung	Gültig ab
1	Tonnensammlung	01.01.04
2	Sacksammlung	01.01.04
3	Depotcontainersammlung	01.01.04
4	Bündelsammlung	01.01.04
5	Wertstoffhofsammlung	01.01.04
6	Korbsammlung	01.01.04
7	Wertstoffmobilsammlung	01.01.04

11.4.26. Liste Systembetreiber

Kürzel	Bezeichnung	Gültig ab
58419	DSD	01.01.2004
50115	Landbell	01.10.2003
55794	ÖRE	01.01.2004
63711	Interseroh	01.01.2004
80228	EKO-PUNKT	01.09.2006

11.5. Formate

Die Import- bzw. Exportdefinitionen (d.h. die Dateien -interface.xml) erlauben jeweils die Angabe von *Formaten* als Attribute des XML-Elements *variable*.

Beispiel:

```
<variable ref="DATUM" field="!!SPALTENNAME!!" format="dd.MM.yy"/>
```

Für den Export wird hierüber die Umwandlung des Wertes in seine textliche Darstellung gesteuert. Umgekehrt definieren diese Attribute die Umwandlung der textlichen Darstellung in einen Wert beim Import.

Dieser Abschnitt ist als kurze Referenz für die möglichen Angaben im *format* Attribut gedacht. wme.fact baut in weiten Teilen auf die in Java standardmäßig vorhandenen Formatter (Package *java.text*) auf, daher ist dies hier größtenteils eine Wiedergabe dessen, was in der „Java 2 Platform, Standard Edition, v 1.3.1 API Specification“ von Sun Microsystems zu finden ist.

11.5.1. Datums- und Zeitangaben

Die möglichen Formatangaben für Datums- und Zeitfelder entsprechen exakt der in der Javadoc zu *java.text.SimpleDateFormat* definierten Syntax. Hier ein Zitat des relevanten Abschnitts:

To specify the time format use a time pattern string. In this pattern, all ASCII letters are reserved as pattern letters, which are defined as the following:

Symbol	Meaning	Presentation	Example
G	era designator	(Text)	AD
y	year	(Number)	1996
M	month in year	(Text & Number)	July & 07
d	day in month	(Number)	10
h	hour in am/pm (1~12)	(Number)	12
H	hour in day (0~23)	(Number)	0
m	minute in hour	(Number)	30
s	second in minute	(Number)	55
S	millisecond	(Number)	978
E	day in week	(Text)	Tuesday
D	day in year	(Number)	189
F	day of week in month	(Number)	2 (2 nd Wed in July)
w	week in year	(Number)	27
W	week in month	(Number)	2
a	am/pm marker	(Text)	PM
k	hour in day (1~24)	(Number)	24
K	hour in am/pm (0~11)	(Number)	0
z	time zone	(Text)	Pacific Standard Time
'	escape for text	(Delimiter)	
"	single quote	(Literal)	'

The count of pattern letters determines the format.

- (Text): 4 or more pattern letters—use full form, < 4--use short or abbreviated form if one exists.
- (Number): the minimum number of digits. Shorter numbers are zero-padded to this amount. Year is handled specially; that is, if the count of 'y' is 2, the Year will be truncated to 2 digits.
- (Text & Number): 3 or over, use text, otherwise use number.
-

Any characters in the pattern that are not in the ranges of ['a'..'z'] and ['A'..'Z'] will be treated as quoted text. For instance, characters like ':', ',', ''', '#', and '@' will appear in the resulting time text even they are not embraced within single quotes.

11.5.2. Zahlen

Die Syntax der Formatangabe für Zahlenfelder lautet:

#,##0.0#####/,/.
#,##0.0#####/./,

^{wm}e.fact Standarddefinition für deutschsprachige Dezimalzahlen
... das gleiche für Englisch