

Briefe schreiben in ADITO

AID 083 DE



© 2015 ADITO Software GmbH

Diese Unterlagen wurden mit größtmöglicher Sorgfalt hergestellt. Dennoch kann für Fehler in den Beschreibungen und Erklärungen keine Haftung übernommen werden. Wir sind für Feedback zu den Themen, Inhalten, aber auch noch vorhandenen Fehlern dankbar und würden uns freuen, Ihre Meinung zu hören. Die in diesen Unterlagen enthaltenen Daten und Angaben, einschließlich URLs und anderer Verweise können ohne vorherige Ankündigung geändert werden. Alle in diesen Unterlagen aufgeführten Produkt- und Firmennamen sind unter Umständen Marken oder geschützte Zeichen der einzelnen Firmen. Ohne ausdrückliche schriftliche Einverständniserklärung der ADITO Software GmbH darf kein Teil dieses Dokumentes vervielfältigt oder in einer Datenverarbeitungsanlage gespeichert oder in diese eingelesen werden. Diese Einschränkung gilt unabhängig von Art und Weise der Datenerfassung.

Autor: FA, MW. Version 10.1. Zuletzt geändert 02.06.2017

Version	Bemerkung
10.1	Trennung des Dokumentes (E-Mail in eigenes Dokument) Anpassung der Platzhalter
10.0	Anpassung an ADITO4.6
2.2	Letzte Version vor Umstellung der Versionshistorie im Dokument

Inhaltsverzeichnis

1.	Überblick	4
1.1.	Briefe	4
1.1.1.	Einzelbriefe	4
1.1.2.	Serienbriefe	4
1.2.	Einzelbriefe	5
1.3.	Das Verfassen eines Einzelbriefs läuft nach folgendem Muster ab:	6
1.3.1.	Einzelbriefe mit XML	6
1.3.2.	Einzelbriefe mit ODT	6
1.3.3.	Einzelbriefe mit Word	8
1.4.	Serienbriefe	10
1.5.	Das Erstellen von Serienbriefen läuft nach dem folgenden Muster ab:	10
1.5.1.	Serienbriefe mit ODT	10
1.5.2.	Serienbriefe mit DOC und DOCX	10
2.	Platzhalter	13
2.1.	Platzhalter ODT	13
2.2.	Platzhalter DOC	13
2.3.	Platzhalter DOCX	13
3.	Exportfields	14
3.1.	Typen von Exportfields	14
3.2.	Adressformate	16
3.3.	Sonderfall Exportfields / Platzhalter in Microsoft Word	16

1. Überblick



Dieses Dokument behandelt die Platzhalter für Dokumente ab dem Referenzsystem 1707. Für eine vorherige Version dieses Dokumentes wenden Sie sich bitte an ADITO.

1.1. Briefe

Ab dem ADITO Referenzsystem 2013, welches mit ADITO4.1 ausgeliefert wird, können Einzel- und Serienbriefe versendet werden. Der Briefversand basiert auf der im Referenzsystem enthaltenen Kontaktmanagement- und Adresslogik.

1.1.1. Einzelbriefe

Einzelbriefe können in den folgenden Formaten verfasst werden:

- OpenDocument-Formate mit direkter Bearbeitung der im ODT liegenden XML-Struktur (ODT)
- DOCX für Microsoft Word mit direkter Bearbeitung der im DOCX liegenden XML-Struktur (XML)
- DOCX und DOC für Microsoft Word unter Verwendung des Microsoft Windows Script Hosts (RunScript)

1.1.2. Serienbriefe

Serienbriefe können in den folgenden Formaten verfasst werden:

- OpenDocument-Formate (ODT) mit direkter Bearbeitung der im ODT liegenden XML-Struktur (ODT)
- DOC bzw. DOCX als Seriendruckdokument mit externer Datenquelle als CSV (Word)

1.1.2.1. Für die Adresslogik verwendete Funktionsbibliotheken, Schlüsselwörter und Attribute innerhalb des Referenzsystems

Funktionsbibliotheken

- lib_wordbrf (Erstellen der Dokumente)
- lib_document (Verwalten der Dokumentvorlagen)
- lib_addrexp (Adresseexport)

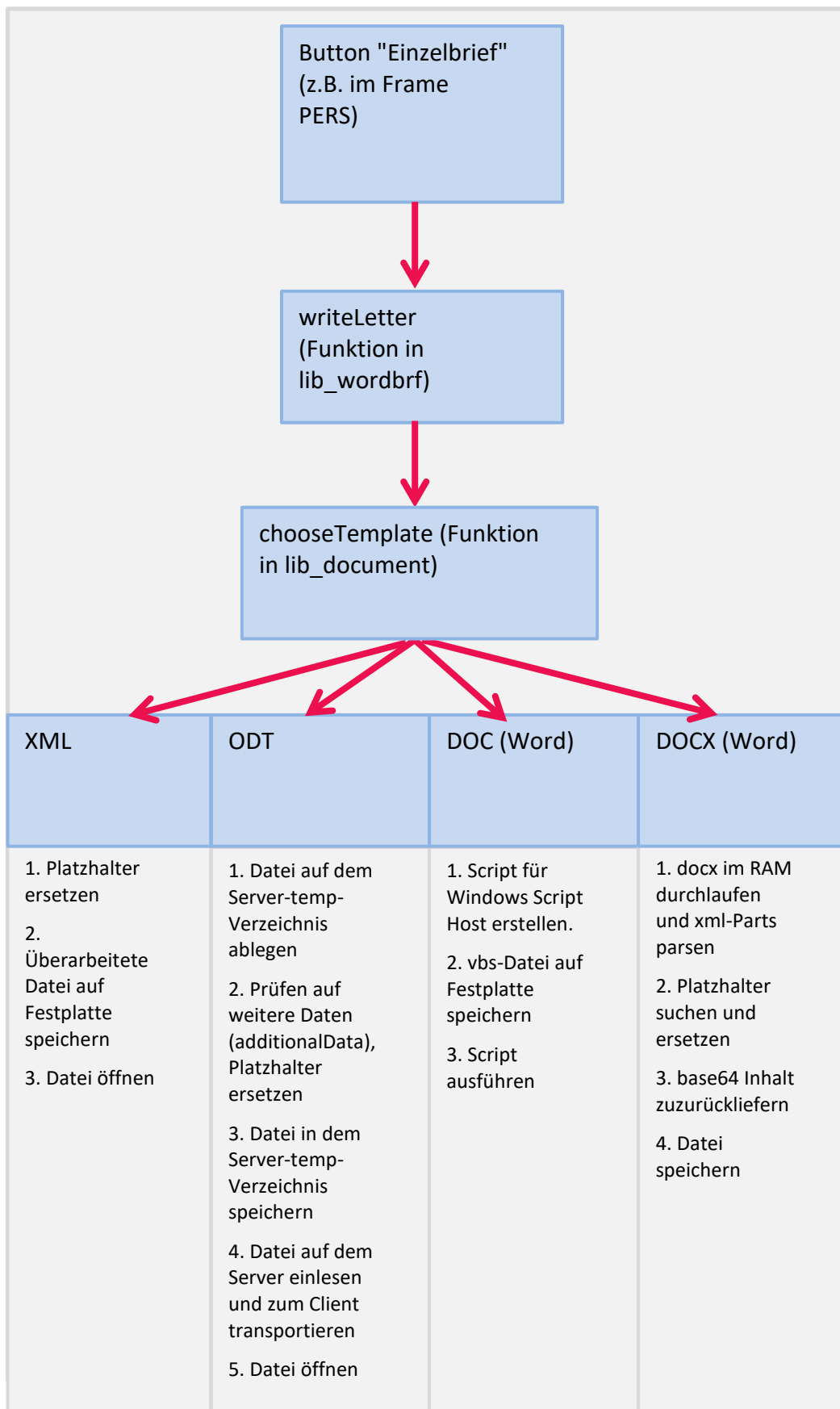
Schlüsselwörter

- Exportfields (Verwaltung der Platzhalter)

Attribute

- zusätzliche Daten / Tabellen Daten (für die Bearbeitung von Tabellen innerhalb von ODT-Dokumenten)
- zusätzliche Daten / weitere Daten (für das Hinzufügen weiterer Daten)

1.2. Einzelbriefe



1.3. Das Verfassen eines Einzelbriefs läuft nach folgendem Muster ab:

1.3.1. Einzelbriefe mit XML

Hierbei handelt es sich um die Verarbeitung der Dokumente vom Typ .docx, welche aber als .xml gespeichert werden. Somit kann man Dateien direkt bearbeiten, ohne dass der Aufruf eines Windows Scripts notwendig ist. Das kann z.B. dann notwendig sein, wenn der Windows Script Host nicht verwendet wird.

1.3.2. Einzelbriefe mit ODT

Eine ODT-Datei ist im Kern eine Zip-Datei. In dieser sind die Style-Informationen und die Textinhalte gespeichert.

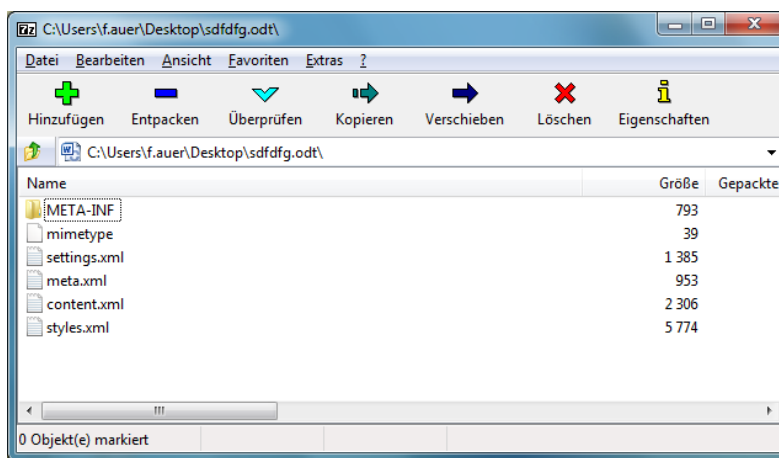


Abbildung: Inhalt einer ODT-Datei, mit dem Programm 7zip geöffnet.

1.3.2.1. Technischer Ablauf

1. **Dokumentvorlage auslesen.** Die ODT-Dokumentvorlage, welche als „Einzelbrief“-Dokumentvorlage mit der Dateierdung .odt gespeichert wurde, wird auf die Server-Festplatte gespeichert.
2. **Datei intern öffnen und content.xml parsen.** Die im ODT-Dokument enthaltene XML-Datei content.xml wird ausgelesen und die Platzhalter in der XML-Struktur ersetzt. Dies geschieht in der Funktion replaceODTFile in der Funktionsbibliothek lib_document. Dabei wird auch TableData und AdditionalData eingefügt.
3. **Datei wieder komprimieren.** Nach dem Überarbeiten wird die XML-Struktur wieder in die Zip-Datei eingefügt und am Server gespeichert.
4. **Datei an Client übergeben.** Die Datei aus dem Server-Verzeichnis geladen und auf der Server-Festplatte gelöscht. Die zuvor eingelesene Datei wird als base64-String an den Client gegeben.

1.3.2.2. AdditionalData auslesen

Reichen die bestehenden Platzhalter aus dem Exportfields-Schlüsselwort nicht aus, so können einem Dokument weitere Daten angefügt werden, ohne dass das Schlüsselwort mit

Hilfe des Designers angepasst werden muss. Dazu wird das Attribut „zusätzliche Daten“ verwendet:

Zusätzliche Daten → Tabellen Daten: Wird verwendet, um Daten in Tabellen zu füllen.

Zusätzliche Daten → weitere Daten: Bietet die Möglichkeit, zusätzliche Platzhalter nur für dieses Dokument einzufügen.

Diese zusätzlichen Platzhalter können wie normale Platzhalter mit @Platzhaltername@ in den Briefen adressiert werden.

Derzeit werden die zusätzlichen Daten nur in **ODT**-Dateien ausgelesen.

Zusätzliche Daten

Der Aufbau des Inhalts in zusätzliche Daten kann wie folgt aussehen:

```
Fields:["RELID", „xx“, „YY"]
Data:[
["RelID", „WertXX“, „WertYY“]
];
```

Unter „Fields“ kann angegeben werden, um welches Feld es sich handelt, wobei das erste immer die RELATIONID ist, auf welche sich der Adress- bzw. Kontaktstammdatensatz bezieht. Im Data-Array kann tabellarisch eine Liste von Fields-Werten angegeben werden, die dann im Dokument ersetzt werden.

1.3.2.3. Anstelle des Data-Arrays kann aber auch ein SQL-String angegeben werden

```
Fields:["RELID", „xx“, „YY"]
SQLStr: „SELECT RELATIONID, FIRSTNAME, LASTNAME FROM RELATION JOIN PERS ON PERS_ID = PERSID“
```

Es können auch Datumswerte formatiert werden:

```
Fields:["RELID", „xx“, „YY“, ["zz", „date“, „form“]]
Data:[
["RelID", „WertXX“, „WertYY“]
]
```

„zz“ gibt hier den Namen für ZZ an, „date“ gibt an, dass es sich um einen Datumswert handelt, und „form“ das entsprechende Datumsformat.

1.3.2.4. Formatierungen können angewandt werden auf

- **date**: Datumswerte
- **long**: Ganzzahlen
- **double**: Kommazahlen

Die Formatierungen entsprechen den in ADITO gültigen Formatmustervorlagen.

Tabellendaten

Die Tabellendaten werden ähnlich angegeben wie die zusätzlichen Daten, nur dass neben dem Namen der Platzhalter auch noch der Name der Tabelle angegeben wird, welche im Dokument existiert. Diese können beispielsweise so aussehen:

```
Table: „ADDR“,
Fields: [„RELID“, „Type“, „Strasse“, „PLZ“, „Ort“, „Staat“, „Land“],
SQLStr: „select RELATION_ID, „ + getKeySQL( „AdressTyp“,
„ADDR_TYPE“) + „ , „ + concat([„ADDRESS“, „BUILDINGNO“]) + „, ZIP,
CITY, STATE, NAME_DE from ADDRESS join COUNTRYINFO on COUNTRY =
ISO2“,
ID: „RELATION_ID“
```

In „Table“ wird der Name der Tabelle im Dokument angegeben.

1.3.2.5. Verwendung im Client

Die zusätzlichen Daten werden im Client in den **Eigenschaften** der Dokumentvorlagen angegeben:

Dokumente Eigenschaften Rechte Admin		
1. Attribut	2. Attribut	Wert
Doku Vorlagen Verwendung		PROPERTY
zusätzliche Daten	weitere Daten	Fields: [„RELID“, „Objekt-Name“], SQLStr: "select '{RELID}', PROPERTY.TITLE " + " from {frameName}" + " where {frameIDColumn} = '{rowID}'"

1.3.3. Einzelbriefe mit Word

Einzelbriefe mit Microsoft Word können sowohl DOC, als auch DOCX-Dateien verarbeiten, je nach installierter Word-Version. Die Aufbereitung erfolgt mit Hilfe einer VBS Datei (Visual Basic Script).

1.3.3.1. Technischer Ablauf - DOC

1. **Dokumentvorlage auslesen.** Die Datei, welche hinter der „Einzelbrief“-Dokumentvorlage liegt, wird ausgelesen und auf die lokale Festplatte des Clients gespeichert.
2. **Script auf Dokumentvorlage anwenden.** Um nun die gewünschten Platzhalter in das Dokument einzufügen, wird ein Windows Script File in der Sprache VBS erstellt. Dieses Script wird ebenfalls auf der Festplatte gespeichert und ausgelesen. Es kann, abhängig von der Dokumentvorlage, so aussehen:

```
set obj = createobject(„Word.Application“)
obj.Visible = true
dim fname
```



```
fname = „C:\Users\f.auer\.adito4\temp\client/e0228151-857b-4125-92ea-1f3eaf25f553/de_Standard.doc“  
obj.Documents.Open fname  
set sel = obj.Selection  
sel.Find.ClearFormatting  
sel.Find.Replacement.ClearFormatting  
dim txt  
  
sel.Find.Execute „@Firmenname 1.Zeile@“, false, false, false, false,  
false, true, 1, false, „Jonny“, 2, false, false, false, false  
[Weitere Überarbeitung]
```

3. Das bestehende Dokument aus der Dokumentenvorlage wird also, einmal gespeichert, über das Skript bearbeitet. Beim Speichern des Historieneintrags wird das überarbeitete Dokument in die Dokumentenmappe übernommen.

1.3.3.2. Technischer Ablauf - DOCX

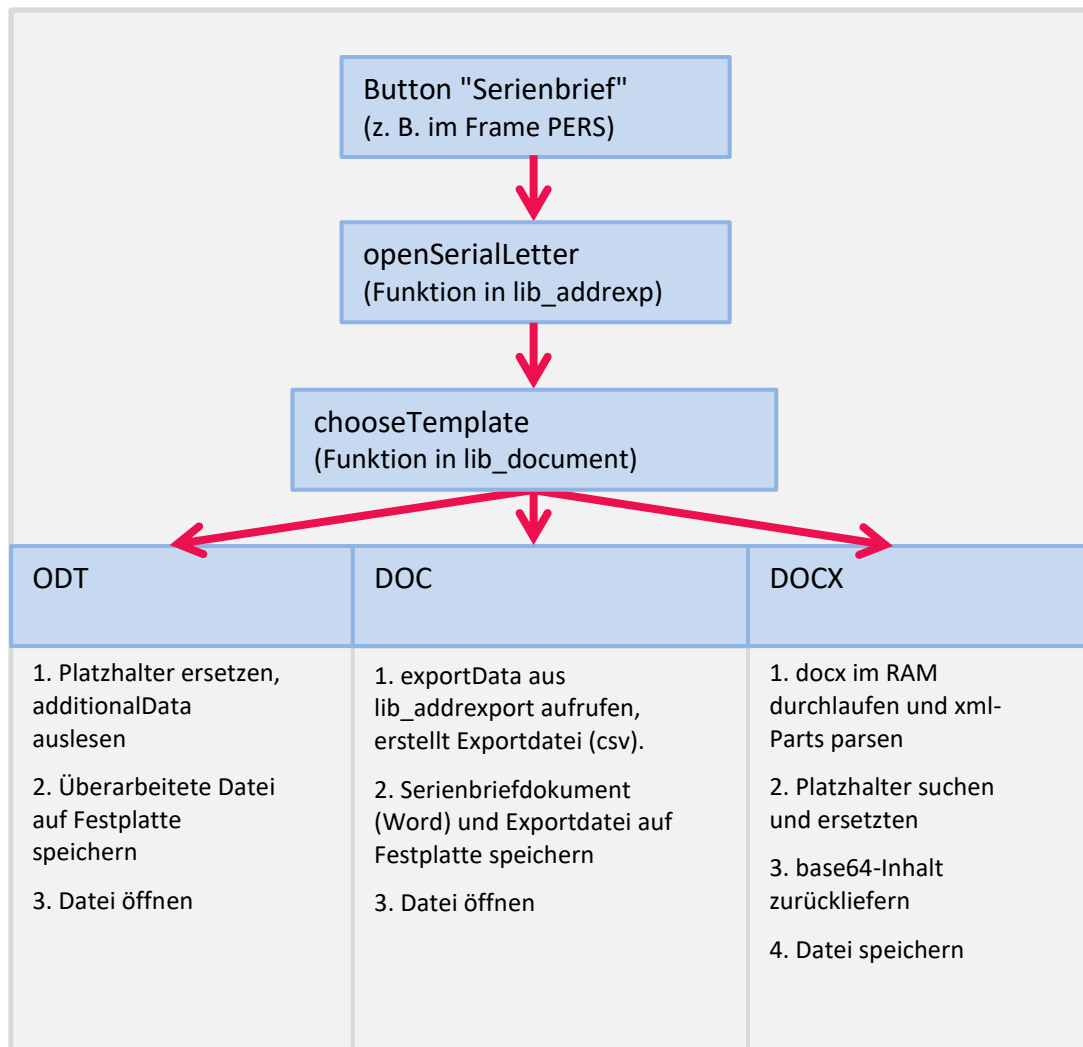
1. **Dokumentvorlage auslesen.** Die Datei, welche hinter der „Einzelbrief“-Dokumentvorlage liegt, wird ausgelesen und das Dokument, sowie alle Platzhalterinformationen werden an den docxtemplater übergeben:
2. **docxtemplater auf Dokumentvorlage anwenden.** Um nun die gewünschten Platzhalter in das Dokument einzufügen, wird im docxtemplater die Datei im RAM entpackt und die benötigten xml-Dateien geparsed. Dann werden die Platzhalter gesucht und mit Hilfe ihrer jeweiligen Funktion durch einen Wert ersetzt.



Wie die Platzhalter verwendet werden, sehen Sie im Kapitel **Fehler! V erweisquelle konnte nicht gefunden werden.**

3. Das Dokument wird anschließend wieder vom docxtemplater gepackt und zurückgegeben. Danach wird die Datei an den Client gegeben.

1.4. Serienbriefe



1.5. Das Erstellen von Serienbriefen läuft nach dem folgenden Muster ab:

1.5.1. Serienbriefe mit ODT

Serienbriefe in ODT funktionieren wie die entsprechenden Einzelbriefe in diesem Format. Für die Serienbriefe wird jeder Brief im selben Dokument kopiert und im XML angefügt.

Die Dokumentvorlagen sind zu Pflegen wie ODT-Einzelbriefvorlagen. Wählen Sie allerdings im Client in der Maske Doku-Vorlagen den Vorlagentyp Exportvorlage aus, können Sie auch die Exportfields auswählen, Exportfields werden allerdings in den ODT-Serienbriefen nicht ausgewertet.

1.5.2. Serienbriefe mit DOC und DOCX

Wird die Serienbrieflogik von Microsoft Word verwendet, dann wird die eigene Seriendruck-Logik von Word verwendet, das heißt:

- Dokumentvorlage ist ein Microsoft Word-Dokument mit aktivierter Seriendruckfunktion.

- Datenquelle ist eine aus der ADITO-Datenbank heraus generierte CSV-Datei.

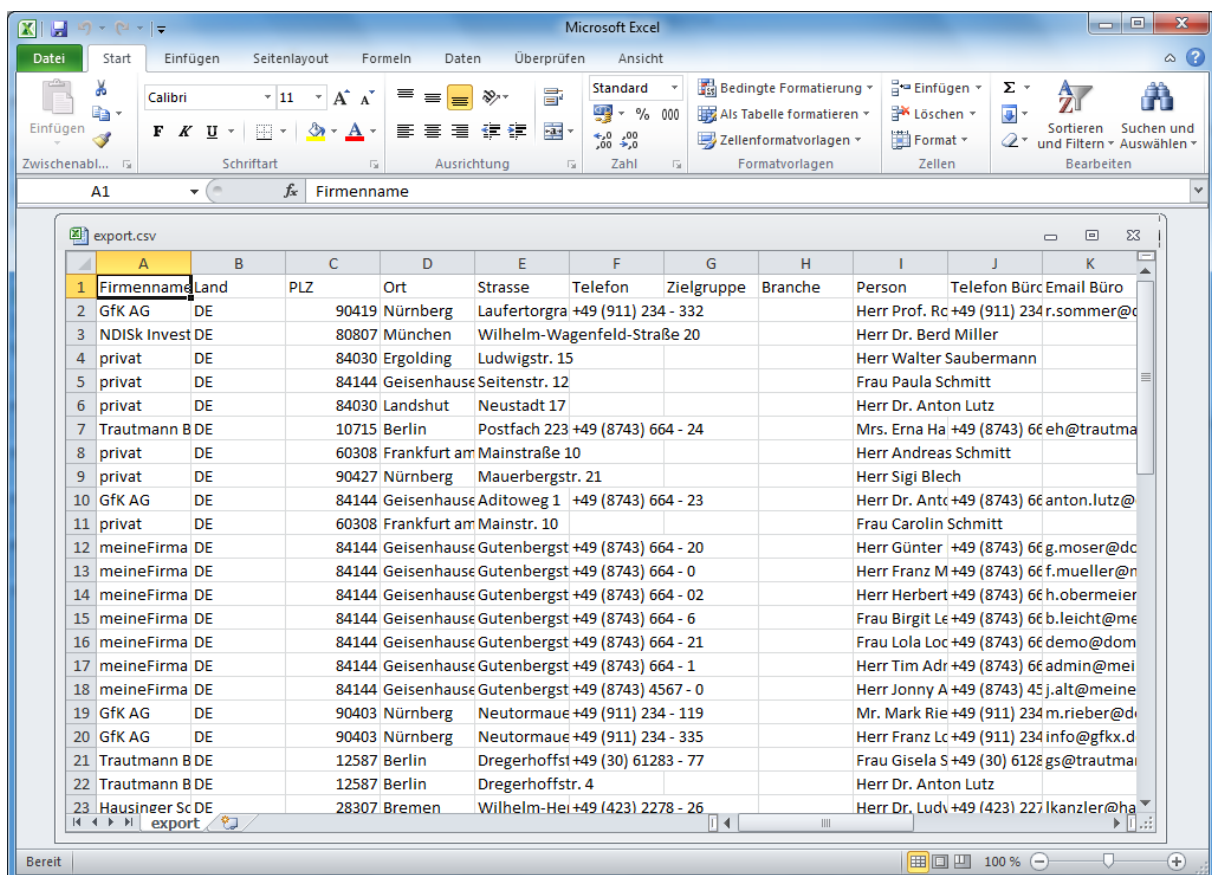
1.5.2.1. Auf Anwenderseite

Folgende Schritte sind durchzuführen, um im Client einen neuen Serienbrief zu erstellen:

1. **Exportvorlage generieren:** Zuerst muss eine Exportdatei generiert werden, welche als Datenquelle für den Seriendruck in Microsoft Word verwendet wird. Dazu eignet sich z.B. die Vorlage „Alle Felder“. Diese erstellt eine CSV-Datei, in welcher alle im Schlüsselwort Exportfields definierten Platzhalter vorhanden sind und füllt diese mit Daten.

Wie viele Daten letztlich exportiert werden, ist dabei egal.

Diese Exportvorlage wird z.B. generiert über Serienaktionen → Export im Frame PERS.



	A	B	C	D	E	F	G	H	I	J	K
	Firmennamen	Land	PLZ	Ort	Strasse	Telefon	Zielgruppe	Branche	Person	Telefon Büro	Email Büro
1	GfK AG	DE	90419	Nürnberg	Laufertorgr	+49 (911) 234 - 332			Herr Prof. Rc	+49 (911) 234 r.sommer@c	
2	NDISK Invest	DE	80807	München	Wilhelm-Wagenfeld-Straße 20				Herr Dr. Bernd Miller		
3	privat	DE	84030	Ergolding	Ludwigstr. 15				Herr Walter Saubermann		
4	privat	DE	84144	Geisenhause	Seitenstr. 12				Frau Paula Schmitt		
5	privat	DE	84030	Landshut	Neustadt 17				Herr Dr. Anton Lutz		
6	Trautmann B	DE	10715	Berlin	Postfach 223	+49 (8743) 664 - 24			Mrs. Erna Ha	+49 (8743) 66 eh@trautma	
7	privat	DE	60308	Frankfurt am	Mainstraße 10				Herr Andreas Schmitt		
8	privat	DE	90427	Nürnberg	Mauerbergstr. 21				Herr Sigi Blech		
9	GfK AG	DE	84144	Geisenhause	Aditoweg 1	+49 (8743) 664 - 23			Herr Dr. Antc	+49 (8743) 66 anton.lutz@	
10	privat	DE	60308	Frankfurt am	Mainstr. 10				Frau Carolin Schmitt		
11	meineFirma	DE	84144	Geisenhause	Gutenbergst	+49 (8743) 664 - 20			Herr Günter	+49 (8743) 66 g.moser@dc	
12	meineFirma	DE	84144	Geisenhause	Gutenbergst	+49 (8743) 664 - 0			Herr Franz M	+49 (8743) 66 f.mueller@n	
13	meineFirma	DE	84144	Geisenhause	Gutenbergst	+49 (8743) 664 - 02			Herr Herbert	+49 (8743) 66 h.obermeier	
14	meineFirma	DE	84144	Geisenhause	Gutenbergst	+49 (8743) 664 - 6			Frau Birgit Le	+49 (8743) 66 b.leicht@me	
15	meineFirma	DE	84144	Geisenhause	Gutenbergst	+49 (8743) 664 - 21			Frau Lola Loc	+49 (8743) 66 demo@dom	
16	meineFirma	DE	84144	Geisenhause	Gutenbergst	+49 (8743) 664 - 1			Herr Tim Adr	+49 (8743) 66 admin@mei	
17	meineFirma	DE	84144	Geisenhause	Gutenbergst	+49 (8743) 4567 - 0			Herr Jonny A	+49 (8743) 45 j.alt@meine	
18	GfK AG	DE	90403	Nürnberg	Neutormaue	+49 (911) 234 - 119			Mr. Mark Rie	+49 (911) 234 m.riever@d	
19	GfK AG	DE	90403	Nürnberg	Neutormaue	+49 (911) 234 - 335			Herr Franz Lc	+49 (911) 234 info@gfkx.d	
20	Trautmann B	DE	12587	Berlin	Dregerhoffst	+49 (30) 61283 - 77			Frau Gisela S	+49 (30) 6128 gs@trautma	
21	Trautmann B	DE	12587	Berlin	Dregerhoffstr. 4				Herr Dr. Anton Lutz		
22	Hausineer Sc	DE	28307	Bremen	Wilhelm-Hei	+49 (423) 2278 - 26			Herr Dr. Ludk	+49 (423) 227 lkanzler@ha	

Abbildung: Ergebnis des Exports

2. **Erstellen des Serienbriefes:** Unabhängig von ADITO wird nun auf Basis dieser bestehenden Exportdatei ein Seriendruckdokument in Word erstellt. Wie es erstellt wird, ist abhängig von der verwendeten Microsoft Word-Version. In Microsoft Word 2010 geschieht dies in einem neuen Dokument über „Sendungen“ → „Vorhandene Liste verwenden. Dort ist die Exportdatei zu wählen, die in Excel betrachtet wurde.

Das Dokument ist dann entsprechend aufzubereiten auf Basis der generierten CSV-Datei. Die Nachfrage nach der Encodierung kann mit „OK“ bestätigt werden.

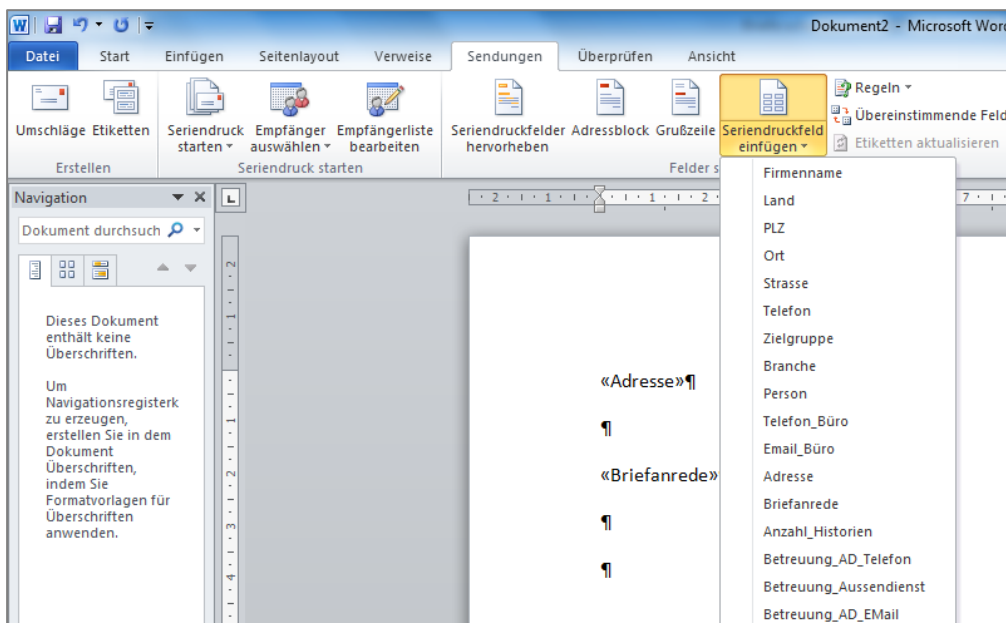


Abbildung: Aufbereiten des Dokumentes.

3. **Erstellung der Dokumentvorlage:** Nun kann innerhalb von ADITO die Dokumentvorlage generiert werden. Dazu im Frame „Doku-Vorlagen“ den Neuanlage-Modus aktivieren und eine neue „Export-Vorlage“ anlegen. Nun kommt der knifflige Teil:
 - a. Unter „Exportfelder“ sind die im Brief verwendeten Felder aus der Exportdatei zu wählen. Exportdateiname, Feld- und Satztrenner sind **genau so** zu wählen, wie es in der Dokumentvorlage des verwendeten Exports definiert ist.
 - b. Unter „Dokument“ muss das vorbereitete Serendruckdokument aus Microsoft Word gespeichert werden. Dabei ist darauf zu achten, dass das Dokument nur geöffnet wurde, während die Vorlage existiert hatte. Ansonsten verliert das Dokument den Serendruckstatus und kann nicht mehr geöffnet werden!
 - c. Die Dokumentvorlage kann gespeichert werden. Über „Serienaktionen“ → „Serienbrief“ im Frame PERS oder jedem anderen Serienbriefbutton im System (z.B. im Verteiler) kann diese Vorlage jetzt verwendet werden. Die ursprüngliche Exportdatei kann gelöscht werden. ADITO legt jetzt bei jedem Export eine neue solche Datei an.

2. Platzhalter

Damit bestimmte Bereiche im Dokument ersetzt werden, werden Platzhalter angegeben. Diese Platzhalter sehen je nach Dateiformat unterschiedlich aus.

2.1. Platzhalter ODT

Platzhalter werden in ODT-Dateien mit {@Platzhaltername@} angegeben. Beispiele: {@Adresse@}, {@Briefanrede@} oder {@Sender E-Mail@}.

2.2. Platzhalter DOC

DOC sollte nicht mehr verwendet werden, die Platzhalter werden mit @@Platzhaltername angegeben, z.B. @@Adresse, @@Briefanrede, @@Sender E-Mail.

2.3. Platzhalter DOCX

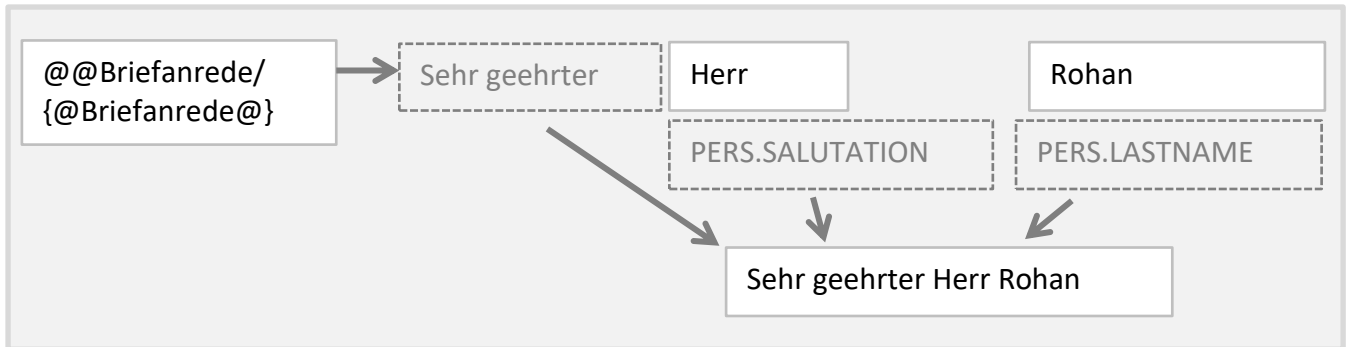
Platzhalter in DOCX werden als einzeilige Platzhalter mit {@einzeiligerPlatzhalter@} und als mehrzeilige Platzhalter mit {@=mehrzeiligerPlatzhalter@} angegeben. Beispiele: {@=Adresse@}, {@Briefanrede@} oder {@=Sender Adresse@}.



Über Customizing können Sie für Platzhaltern weitere Möglichkeiten pflegen.

3. Exportfields

Jede Art von Brief oder E-Mail, die an Personen und Firmen im ADITO-System versendet wird, kann über Exportfields personalisiert werden. Das heißt, dass bestimmte Schlüsselbegriffe vom System durch einen sprechenden Namen ersetzt werden:



Die Daten werden in der Funktionsbibliothek lib_addr zusammengesetzt. Die benötigten Exportfields werden in der aufrufenden Funktion (z.B. lib_wordbrf) aus der Datenbank ausgelesen.

Die Exportfields werden im gleichnamigen Schlüsselwort gepflegt.

3.1. Typen von Exportfields

Die Typen werden in den Details angegeben:

Keyname2	Beschreibung	Beispiel (Details)
fieldname	Bei Erweiterung von Relation, ORG und PERS können alle Tabellenspalten für die übergebene RelationID eingefügt werden. Format: Tabellename .Spaltenname	ORG.ORGNAME //Firmenname übergeben
addressformat	Übliche Adressformatierungsplatzhalter in beliebiger Reihenfolge.	{a}{bn} // Straße Hausnummer
function	Aktuell kann jede Funktion aus lib_keyword oder lib_attr übergeben werden. Der Code wird in der lib_addr mit eval() übergeben und ausgeführt. D. h. sollen Funktionen aus anderen Prozessen verwendet werden,	getCommAddrSQL('Org', 'Telefon', 'ORGREL.RELATIONID') getAttrSQL('Zielgruppe', 'ORGREL.RELATIONID')

Keyname2	Beschreibung	Beispiel (Details)
	<p>dann diese in der lib_addr importieren.</p> <p>Möglich sind zusätzlich alle JDIO Funktionen.</p>	
sender.function	<p>Funktion für den angemeldeten User.</p> <p>Sender lässt sich mit allen Arten Kombinieren. Z.B. sender.fieldname</p>	<pre>getCommAddrSQL('Pers', 'Telefon Büro', 'RELATION.RELATIONID')</pre>
afunction	<p>Über afunction kann zusätzlich zur Funktion von function ein Feld der Oberfläche übergeben werden.</p> <p>Achtung: nur in Einzelbriefen möglich!</p>	<pre>"\$comp.edtANTRAGSNR" var datum = db.cell("select eingangsdatum from antrag where antragid = "" + "\$comp.idcolumn" + ""); datetime.toDate(datum, "dd.MM.yyyy"); //Zur Übergabe einer Adresse die nicht als Standrad markiert ist. z.B. Postfachadresse var relid = db.cell("select relationid from relation where addr_type = 12 and pers_id = '\$comp.persid") new AddrObject("\$comp.idcolumn", relid).getFormattedAddress();</pre>
avalueoffunction	<p>Liest Werte aus einer Komponente im entsprechenden Frame aus.</p>	<pre>\$comp.edtANTRAGSNR</pre>
select	<p>Übergabe eines Subselects, hängt ab von der Tabelle ORG, PERS, RELATION, ADDRESS.</p>	<pre>select count(*) from HISTORYLINK join HISTORY on HISTORYID = HISTORY_ID where ROW_ID = RELATION.RELATIONID and OBJECT_ID in (1, 2, 3)</pre>

3.2. Adressformate

Typ	Wert	Typ	Wert	Typ	Wert
{on}	Orgname, Name der Organisation	{hs}	Headsalutation, Anrede im Briefkopf	{ls}	Lettersalutation, Briefanrede
{sa}	Salutation, Anrede	{ti}	Title, Titel	{dp}	Department, Abteilung
{fn}	Firstname, Vorname	{mn}	Middlename, zweiter Vorname	{ln}	Lastname, Nachname
{al}	Addressline, Straße	{bn}	Building Number, Hausnummer	{zc}	Zipcode, Postleitzahl
{ci}	City, Ort	{di}	District, Landkreis	{rg}	Region
{st}	State, Staat	{cc}	Countrycode, Ländercode	{cn}	Countryname, Länderbezeichnugn

3.3. Sonderfall Exportfields / Platzhalter in Microsoft Word

In Microsoft Word kann es vorkommen, dass Platzhalter nicht ersetzt werden. In diesem Fall kann es sein, dass ein Formatierungsumbruch oder ähnliches in der XML-Datei zwischen den Steuerzeichen (@Platzhaltername@) und dem eigentlichen Platzhalter einen Umbruch vornimmt, weshalb das Auslesen der Platzhalter nicht mehr möglich ist.

Unter Word 2013 kann das Abschalten der Kombiniergenauigkeit diese Unterbrechung verhindern:

DATEI → Optionen → Trust Center (in der Sidebar links) → Button "Einstellungen für das Trust Center...." → Datenschutzooptionen (in der Sidebar links) → Häkchen bei "Zufallszahlen zur Verbesserung der Kombiniergenauigkeit speichern" rausnehmen und dann die Vorlage speichern.