

# Starterframework TONIC

Konzeption und Benutzung

Version: 6.13  
Stand: 01.12.2022

1	Einführung.....	4
1.1	M/TEXT Version.....	4
1.2	Begriffe .....	4
1.2.1	Auftragsteuerung .....	4
1.2.2	Korrespondenzen .....	4
1.2.3	Exemplare .....	4
1.2.4	Brief.....	4
1.2.5	Anschreiben .....	5
1.2.6	Anlage, Beilage, Beileger .....	5
1.2.7	Sendung .....	5
1.2.8	Kopie, Durchschrift .....	5
1.2.9	Usecases.....	6
1.2.9.1	Eine Korrespondenz mit einem Exemplar.....	6
1.2.9.2	Eine Korrespondenz mit einem Erst-Exemplar und weiteren Exemplaren .....	6
1.2.9.3	Zwei Korrespondenzen mit jeweils einem Exemplar.....	7
1.3	Weitere Begriffe.....	7
1.3.1	Empfänger und alternativer Empfänger .....	7
1.3.2	Absender .....	7
1.3.3	Usecase .....	8
1.3.3.1	Erst-Exemplar mit einem alternativen Empfänger.....	8
1.4	Weitere Begriffe.....	8
1.4.1	Teildokumente – die Bestandteile eines Exemplars.....	8
1.4.2	Teildokumenttypen.....	8
1.4.2.1	Usecase: Unterdrücken von Teildokumenten bei bestimmten Exemplaren .....	9
1.4.3	Ausgabekanäle .....	9
1.4.4	Ausgabesteuerung.....	9
1.4.5	Ausgabeziel .....	9
2	Datenmodelle .....	10
2.1	Auftragssteuerung.datamodel .....	10
2.2	Korrespondenz.datamodel .....	11
2.3	Exemplar.datamodel .....	12
2.4	Datenmodelle Absender und Empfaenger .....	13
2.5	Ausgabesteuerung.datamodel .....	13
2.6	Datenmapping: Anforderungen an die Datenquelle.....	13
3	Aufgaben des Frameworks .....	14
3.1	Exkurs Inhaltserweiterungen .....	14
3.2	Inhaltserweiterungen im Framework.....	16

3.3	Auftragssteuerung: Korrespondenzen und Exemplare .....	18
3.3.1	Korrespondenz-Baustein .....	18
3.3.2	Alle Exemplare in der M/Workbench anzeigen .....	18
4	Beispiel-Teildokumenttypen .....	19
4.1	Allgemeine Layoutelemente .....	20
4.2	Brief-Teildokument .....	20
4.2.1	Frameworkbausteine im Brief-Teildokument.....	22
4.3	AGB-Teildokument .....	23
4.4	Anlage-Teildokument.....	25
4.5	Infoblatt-Teildokument.....	27
4.6	Rückantwort-Teildokument.....	29
5	Styles.....	31
6	Beispielvorlagen .....	32
7	M/OMS Ansteuerung .....	33
7.1	Baustein Ausgabesteuerung .....	33
7.1.1	ZENTRALDRUCK .....	33
7.1.2	ARCHIV.....	33
7.1.3	LOKALDRUCK.....	34
7.1.4	FAX .....	34
7.1.5	EMAIL oder EMAILATTACHMENT.....	35
7.2	M/OMS Konfiguration.....	36
7.2.1	ZENTRALDRUCK-Stapel .....	37
7.2.2	ZENTRALDRUCK_PDFVT-Stapel .....	38
7.2.3	ARCHIV-Stapel.....	39
7.2.4	EMAIL-Stapel.....	40
7.2.5	LOKALDRUCK-Stapel.....	41
8	Adaption des Frameworks an Kundenprojekte .....	41

# 1 Einführung

Das Starterframework TONIC ist eine Beispielimplementierung eines Frameworks zur Dokumentenerzeugung und Dokumentensteuerung mit M/TEXT TONIC. Es soll und muss an die kundenspezifischen Anforderungen angepasst werden. Das Starterframework zeigt in seiner Implementierung wie bestimmte Funktionen mit TONIC umgesetzt werden können.

Das Starterframework ist mit seinen Grundfunktionen im Projekt **Framework\_DE** implementiert.

Im Projekt **Configuration** ist eine beispielhafte Umsetzung einer M/OMS Ansteuerung zu finden.

Das Projekt **Framework\_Demo\_DE** zeigt Beispielvorgaben, welche die Grundfunktionen der Dokumentensteuerung beinhalten. Auf einige dieser Beispielvorgaben wird in diesem Dokument zur Veranschaulichung referenziert.

**Dieses Dokument setzt Kenntnis zu M/TEXT TONIC und M/OMS voraus und ersetzt nicht die vorhandenen Produkthandbücher.**

Im nachfolgenden werden die benutzten Begriffe erklärt und das Konzept des Starterframeworks veranschaulicht.

## 1.1 M/TEXT Version

Die minimal benötigte Version M/Text-Version für das vorliegende Framework kann in der Datei Framework\_DE\MText\_Version.txt nachgeschaut werden.

## 1.2 Begriffe

### 1.2.1 Auftragsteuerung

Die Anforderung an die Serie M/ ein oder mehrere Dokumente für den Versand oder die Ablage zu erstellen, bezeichnet das Starterframework als Auftragssteuerung. Die Auftragsteuerung umfasst alle Informationen, die zur Erstellung der angeforderten Dokumente notwendig ist.

Eine Dokumentvorlage benötigt mindestens das Auftragssteuerungs-Datenmodell. Zusätzlich kann eine Vorlage ein Fachdatenmodell einbinden. Das Ergebnis der Dokumentvorlage ist technisch ein M/TEXT-Dokument, das durch M/TEXT erstellt und mittels M/OMS verarbeitet wird.

### 1.2.2 Korrespondenzen

Eine Korrespondenz ist eine Nachricht mit einem bestimmten Inhalt und mindestens einem bestimmten Empfänger. Es können mit einem Datenstrom mehrere Korrespondenzen in einem Vorgang erzeugt werden. Diese haben in der Regel unterschiedliche Inhalte.

### 1.2.3 Exemplare

Ein Exemplar ist eine bestimmte Ausprägung der Korrespondenz, die nach bestimmten Regeln gebildet wird. Es kann mehrere Exemplare einer Korrespondenz geben, die aber alle die gleiche Nachricht beinhalten. Das erste Exemplar ( instance(0) ) ist immer das Original. Weitere Exemplare sind Kopien dieser Korrespondenz, die an andere Empfänger gehen können. Jedes weitere Exemplar hat einen Brief, der dem Original-Brief vorangestellt wird.

### 1.2.4 Brief

Ein Brief bezeichnet einen nach DIN-5008 konformen Geschäftsbriefbogen oder einen solchen nach dem Corporate Design des Kunden.

### 1.2.5 Anschreiben

Ein Anschreiben ist ein Brief der anderen Briefen vorangestellt wird, an einen Empfänger gerichtet, dem ein oder mehrere Briefe folgen können.

### 1.2.6 Anlage, Beilage, Beileger

Diese Begriffe bezeichnen Teile, die einem Brief hinzugefügt werden können. Dies können z. B. PDF-Anlagen, Fragebögen oder AGBs sein.

### 1.2.7 Sendung

Eine Sendung bezeichnet alle Anschreiben, Briefe und Anlagen, die einem Empfänger zugestellt werden.

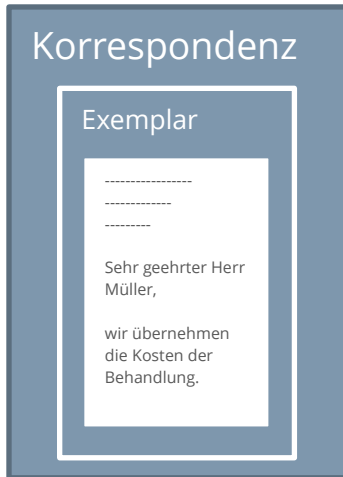
### 1.2.8 Kopie, Durchschrift

Eine Kopie ist ein weiteres Exemplar des Original- oder Erst-Exemplars. Eine Kopie kann an den Empfänger des ersten Exemplars gehen oder aber auch an weitere Empfänger.

## 1.2.9 Usecases

### 1.2.9.1 Eine Korrespondenz mit einem Exemplar

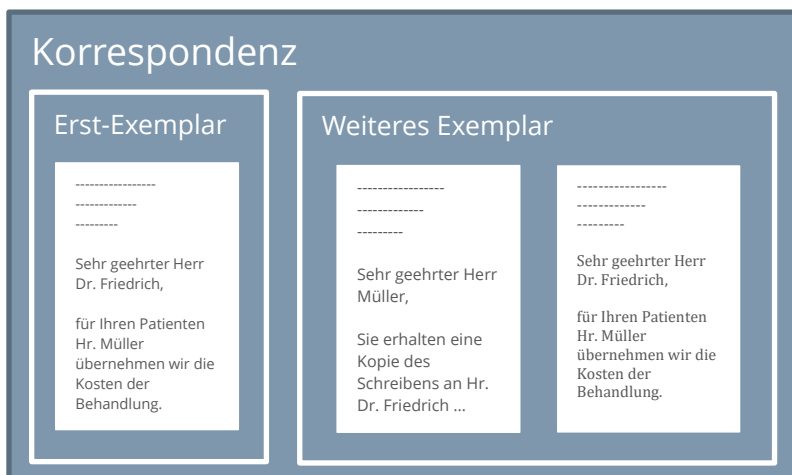
Es wird ein Brief an einen Versicherten verschickt, dass die Kosten einer ärztlichen Behandlung übernommen werden.



Die Beispielvorlage zu diesem Usecase finden Sie im Starterframework im Projekt Framework\_Demo\_DE im Ordner Vorlagen. Wählen Sie dort die Vorlage Demo\_EineKorrespondenz.template aus und darin den Testfall 1Korrespondenz\_1Exemplar\_zentral.

### 1.2.9.2 Eine Korrespondenz mit einem Erst-Exemplar und weiteren Exemplaren

Ein Beispiel für eine Korrespondenz mit einem Erstexemplar und einem weiteren Exemplar ist, wenn der Arzt einen Brief bekommt, in dem ihm mitgeteilt wird, dass die Kosten für die ärztliche Behandlung seines Patienten übernommen werden. Der Patient bekommt den Brief an den Arzt zur Kenntnissnahme in Kopie. Der Arzt ist Empfänger des Erstexemplars. Der Patient ist Empfänger des weiteren Exemplars.

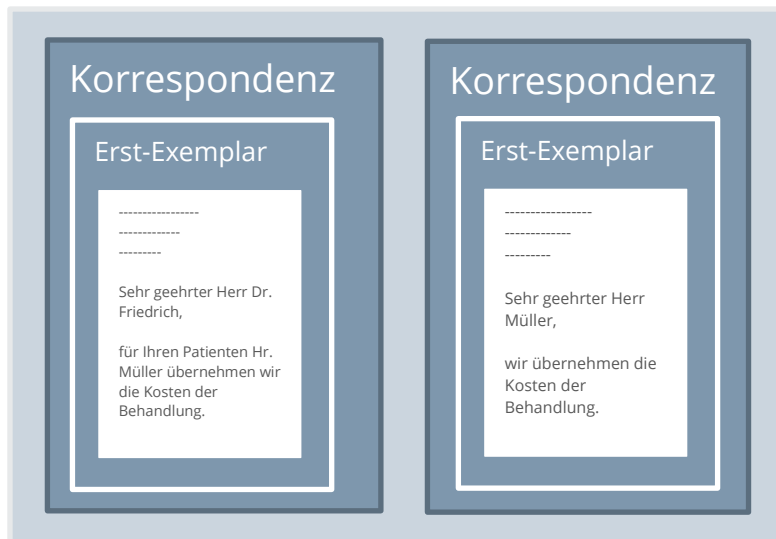


Für dieses Beispiel wählen Sie in der Vorlage Demo\_EineKorrespondenz\_ZweiExemplare.template den Testfall 1Korrespondenz\_2Exemplare\_zentral aus.

Um in der M/Workbench alle Exemplare angezeigt zu bekommen schalten Sie die Exemplarsteuerung um, wie es im Abschnitt 3.3.2 Alle Exemplare in der M/Workbench anzeigen beschrieben ist.

### 1.2.9.3 Zwei Korrespondenzen mit jeweils einem Exemplar

Ein Beispiel für eine Vorlage mit zwei Korrespondenzen ist, bezogen auf das obige Beispiel, wenn in einem Auftrag unabhängig dem Versicherten und dem behandelnden Arzt gleichzeitig je ein Brief geschickt wird. Dem Versicherten und dem Arzt werden jeweils in getrennten Briefen mitgeteilt, dass die Behandlungskosten übernommen werden. Die zwei Korrespondenzen haben also einen unterschiedlichen Inhalt.



Zur Demonstration wählen Sie die Vorlage Demo\_ZweiKorrespondenzen.template mit dem Testfall 2Korrespondenzen\_1Exemplar\_zentral aus.

## 1.3 Weitere Begriffe

### 1.3.1 Empfänger und alternativer Empfänger

Jedes Exemplar hat einen Empfänger. In einem Brief wird die Adresse des Empfängers im Adressfenster des Briefumschlags angedruckt.

Jedes Exemplar kann einen alternativen Empfänger haben. Dieser hat bei den Kunden verschiedene Namen und Bedeutungen, z.B. Betreuer oder Postbevollmächtigter. Der alternative Empfänger ist optional.

Das Starterframework erzeugt bei Vorhandensein eines alternativen Empfängers einen weiteren Brief, der dem eigentlichen Brief vorangestellt wird. Dieser vorangestellte Brief hat einen vordefinierten Inhalt. Beide Briefe liegen hintereinander im Briefkuvert und werden so an den alternativen Empfänger geschickt.

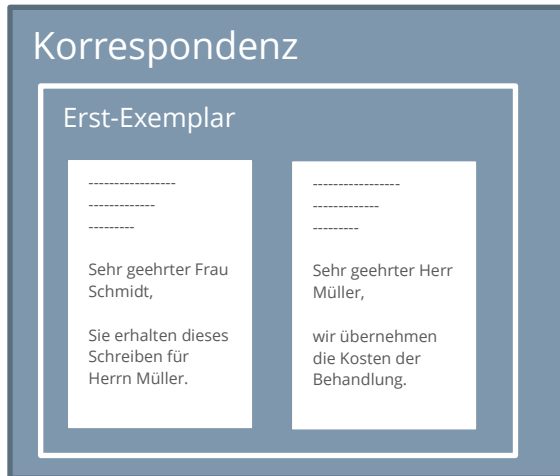
### 1.3.2 Absender

Als Absender wird die Adresse bezeichnet, die in der Regel oberhalb des Adressfensters angedruckt wird. Der Absender ist für alle Exemplare immer identisch, kann aber für verschiedene Korrespondenzen unterschiedlich sein.

### 1.3.3 Usecase

#### 1.3.3.1 Erst-Exemplar mit einem alternativen Empfänger

Wenn ein Versicherter einen Postbevollmächtigten bzw. einen Betreuer hat, dann wird die gesamte Kommunikation über diesen Partner abgewickelt und nicht mit dem Versicherten. Im Adressfenster des Briefumschlags muss die Adresse des Postbevollmächtigten sichtbar sein.



Wählen Sie in die Vorlage Demo\_EineKorrespondenz.template und den Testfall 1Korrespondenz\_1Exemplar\_AlternativerEmpfaenger aus.

## 1.4 Weitere Begriffe

### 1.4.1 Teildokumente – die Bestandteile eines Exemplars

Ein Exemplar kann sich aus mehreren Teildokumenten zusammensetzen. Teildokumente unterscheiden sich durch ihr Layout und ihren Inhalt. Ein Teildokument kann eine oder mehrere Seiten umfassen.

Ein Exemplar kann z. B. aus einem Teildokument Brief bestehen, das von einem Teildokument Infoblatt sowie von Allgemeinen Vertragsbedingungen gefolgt ist. Ein Beispiel dazu finden Sie in der Vorlage Demo\_EineKorrespondenz\_Infoblatt\_AGB.template.

### 1.4.2 Teildokumententypen

Das Layout von Teildokumenten wird in der Regel durch das Corporate Design, die Richtlinien zur Korrespondenzerstellung oder das Marketing vorgegeben. Bestimmte Teildokument werden ein immer wiederkehrend gleiches Layout haben.

Typische Teildokumente mit jeweils gleichem Layout sind z. B.:

- Brief
- Infoblatt
- Versicherungsschein
- AGB

Das Starterframework enthält Beispiele von Teildokumententypen und deren Nutzung in Vorlagen.



#### 1.4.2.1 Usecase: Unterdrücken von Teildokumenten bei bestimmten Exemplaren

Standardmäßig sind alle Teildokumente in allen Exemplaren vorhanden. Es ist aber möglich, bestimmte Teildokumente in bestimmten Exemplaren zu unterdrücken. So könnte z. B. ein Arzt zusätzlich zum Anschreiben ein Infoblatt zur Kostenübernahme bekommen. Das weitere Exemplar, welches an den Patienten geht, benötigt dieses Infoblatt nicht.



Ein Beispiel dazu finden Sie in der Vorlage

Demo\_EineKorrespondenz\_ZweiExemplare\_Infoblatt.template mit dem Testfall 1Korrespondenz\_2Exemplare\_zentral.

Um in der M/Workbench alle Exemplare angezeigt zu bekommen schalten Sie die Exemplarsteuerung um, wie es im Abschnitt 3.3.2 Alle Exemplare in der M/Workbench anzeigen beschrieben ist.

#### 1.4.3 Ausgabekanäle

Ausgabekanäle bestimmen, wie Dokumente an M/OMS übergeben und dort weiterverarbeitet werden. Das Starterframework hat bereits verschiedene Ausgabekanäle definiert, die zeigen, wie eine Übergabe realisiert werden kann. Diese sind:

- ARCHIVE
- EMAIL
- FAX
- LOCALPRINTING
- CENTRALPRINTING

Auch hier gilt, wie für das gesamte Starterframework, dass dies nur Beispiele sind, die an die jeweiligen Bedürfnisse angepasst und erweitert werden müssen.

#### 1.4.4 Ausgabesteuerung

Die Ausgabesteuerung regelt, welche Parameter für einzelne Ausgabeziele gesetzt werden. Dies ist z. B. für das Ausgabeziel EMAIL die E-Mail-Adresse des Empfängers.

#### 1.4.5 Ausgabeziel

Das Ausgabeziel ist der Auszug aus dem M/TEXT-Dokument, der für die Verarbeitung durch einen M/OMS-Stapel erstellt wird. Der Name des Ausgabeziels ist das Medium und entspricht in vielen Fällen dem Ausgabekanal.

- ARCHIVE
- EMAIL
- EMAILATTACHMENT
- FAX
- LOCALPRINTING
- CENTRALPRINTING

Das Ausgabeziel kann unterschiedliche Parameter haben. Die Ausgabeziele werden durch entsprechende M/OMS-Stapel verarbeitet und den Folgeprozessen zur Verfügung gestellt.

## 2 Datenmodelle

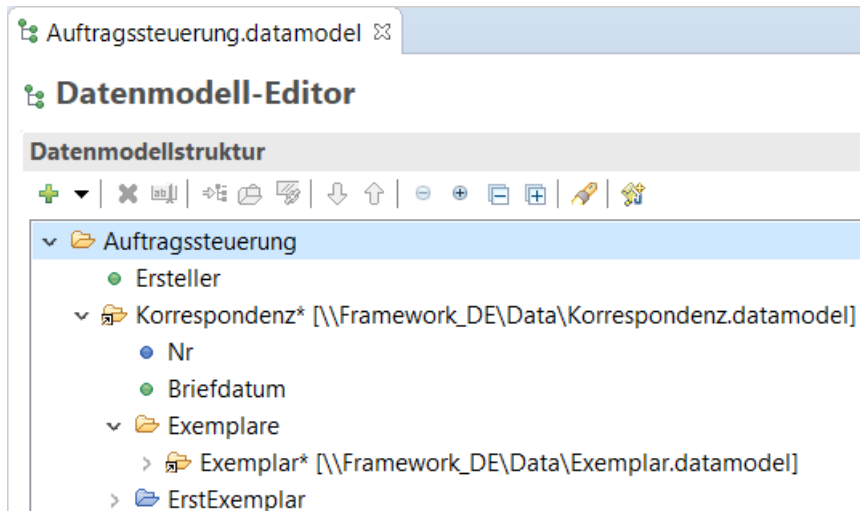
Nachfolgend werden die wichtigsten Datenmodelle beschrieben und deren Aufgaben erläutert. Details zum Aufbau der einzelnen Datenmodelle kann man jederzeit im Framework nachsehen. Alle Framework-Datenmodelle befinden sich im Projekt Framework im Verzeichnis Data.

### 2.1 Auftragssteuerung.datamodel

Grundlage eines Auftrags ist das Datenmodell Auftragssteuerung.

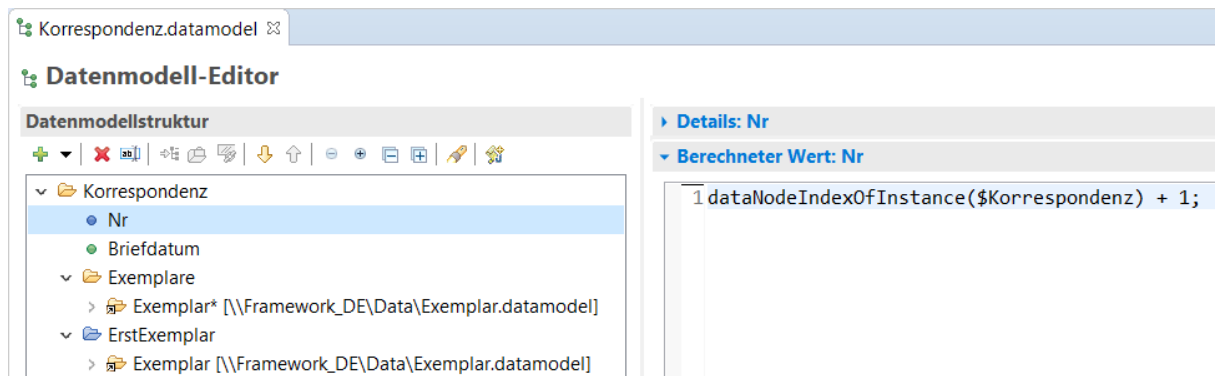
Dieses beinhaltet alle Informationen zur Erstellung der Korrespondenzen mit ihren verschiedenen Exemplaren. Für jedes Exemplar werden Empfänger-, Absender- und Verarbeitungsinformationen im M/OMS gespeichert.

Die Korrespondenzen werden durch den Mehrfachknoten **Korrespondenz\*** dargestellt. Innerhalb des Korrespondenz-Knotens ist der Knoten **Exemplare** zu finden, welcher wiederum den Mehrfachknoten **Exemplar\*** enthält.



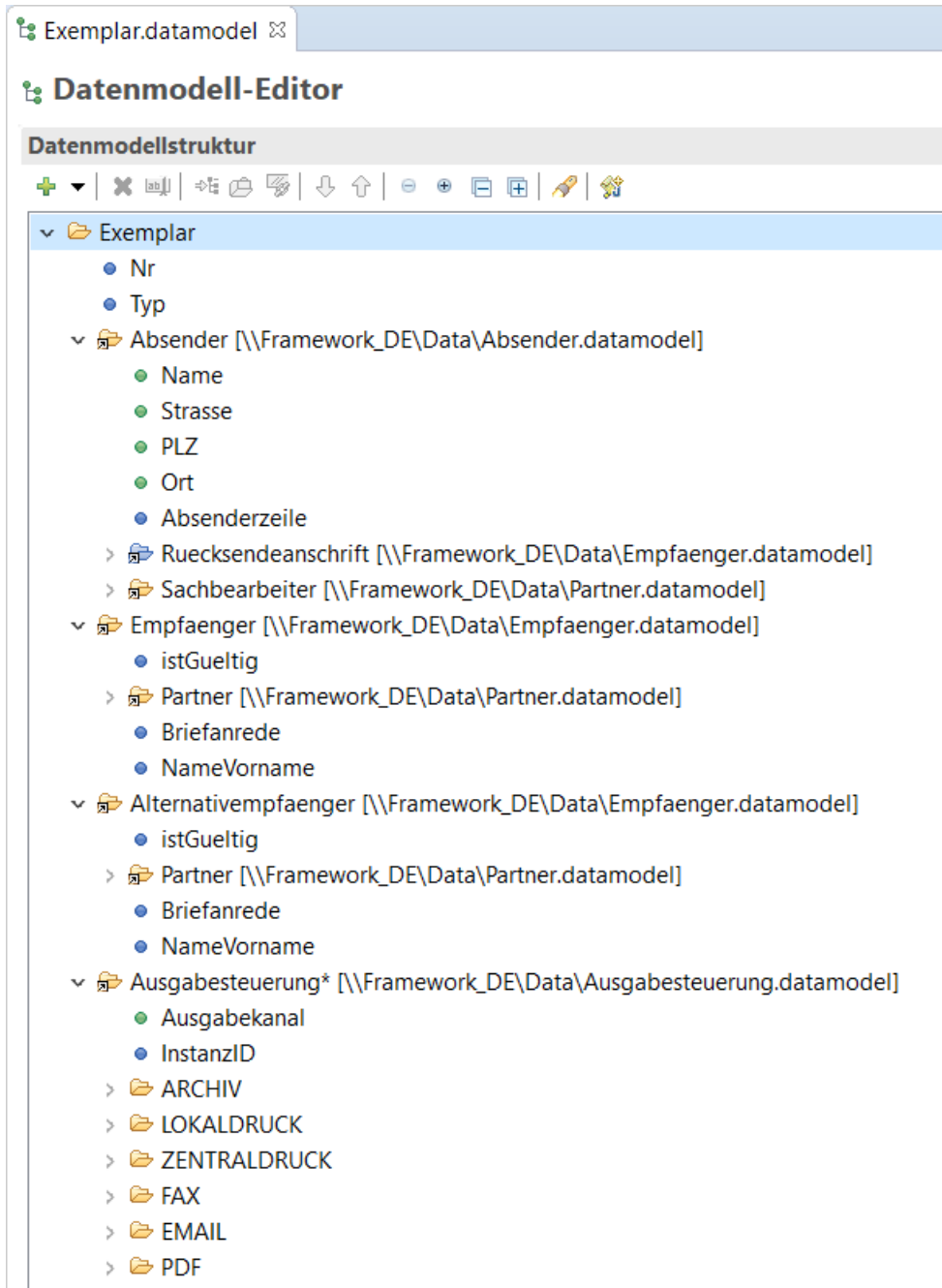
## 2.2 Korrespondenz.datamodel

Das Korrespondenz-Datenmodell enthält alle Exemplare zu einer Korrespondenz. Außerdem wird ein Knoten **Nr** berechnet, der die aktuelle Nummer der Korrespondenz enthält.



## 2.3 Exemplar.datamodel

Der Mehrfachknoten **Exemplar\*** enthält die Informationen zu jeweils einem Exemplar. Dies sind die Unterknoten **Absender**, **Empfaenger**, **Alternativempfaenger** sowie der Mehrfachknoten **Ausgabesteuerung\***.



Außerdem wird der Knoten **Nr** berechnet, der die jeweilige Exemplarnummer innerhalb einer Korrespondenz beinhaltet.

Der berechnete Knoten **Typ** enthält ORIGINAL für das Erstexemplar oder WEITERES\_EXEMPLAR für nachfolgende Exemplare.

## 2.4 Datenmodelle Absender und Empfänger

Diese Datenmodelle werden hier nicht explizit beschrieben da ihre Struktur den kundenindividuellen Gegebenheiten angepasst werden muss.

## 2.5 Ausgabesteuerung.datamodel

Das Ausgabesteuerung-Datenmodell enthält die für die Ausgabekanäle notwendigen Parameter. Die Knoten entsprechen jeweils einem Ausgabekanalnamen. Sie müssen für die Übergabe an M/OMS an die jeweiligen Gegebenheiten angepasst und erweitert werden.

Diese Parameter werden in FW\_Ausgabesteuerung.outputSettings den jeweiligen Ausgabezielen übergeben.

The screenshot shows the 'Datenmodell-Editor' interface. On the left, the 'Datenmodellstruktur' (Data Model Structure) tree is expanded to show the 'Ausgabesteuerung' (Output Control) model. It lists various output channels like ARCHIV, LOKALDRUCK, ZENTRALDRUCK, FAX, EMAIL, and PDF, each with associated parameters like PartnerID, UserID, Druckernamen, Batch, PLZ, LKZ, Absender, Empfänger, and Betreff.

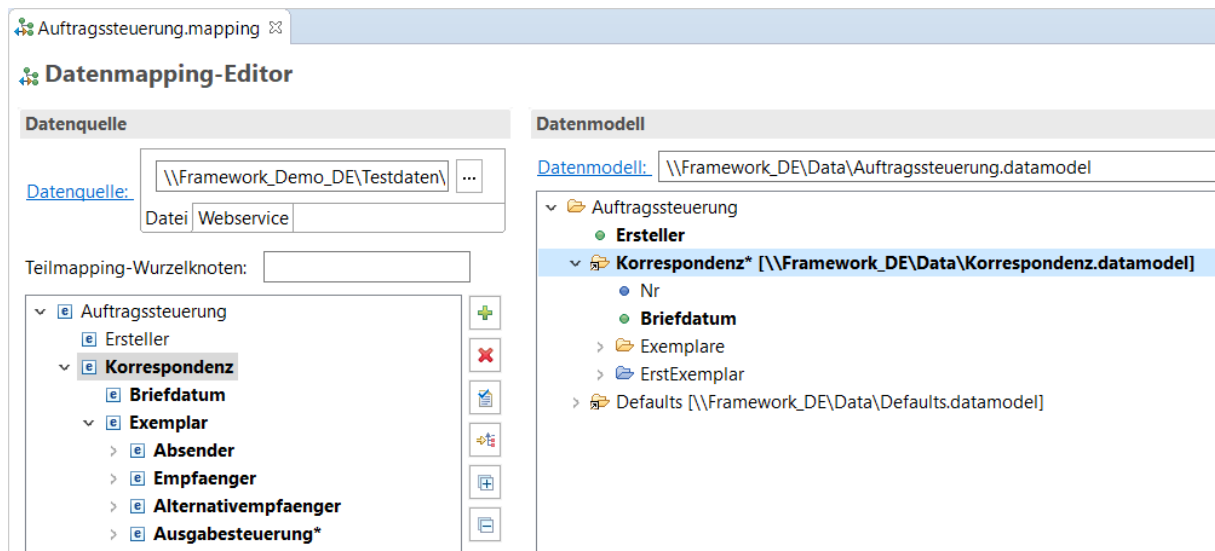
On the right, a detailed view of the 'Ausgabezielparameter' (Output Target Parameters) for the 'EMAIL' target is shown. It lists the following parameters and their values:

Name	Wert
MAIL_FROM	\$Ausgabesteuerung.EMAIL.Absender
MAIL_TO	\$Ausgabesteuerung.EMAIL.Empfänger
MAIL_SUBJECT	\$Ausgabesteuerung.EMAIL.Betreff.isEmpty()

## 2.6 Datenmapping: Anforderungen an die Datenquelle

Das Datenmodell Ausgabesteuerung und damit auch das Ausgabesteuerung-Datenmapping setzt eine Datenquelle (z. B. XML oder JSON) voraus, welche dem Datenmodell entspricht.

Da dies in der Regel nicht zutrifft, ist entweder das Mapping anzupassen oder das Datenmodell um JavaScript-Funktionen zu erweitern, die das Datenmodell korrekt befüllen. Ebenso kann in Betracht gezogen werden, das Datenmodell umzubauen, um der liefernden Datensituation zu entsprechen. In jedem Fall sind individuelle Anpassungen für den Kunden notwendig.



### 3 Aufgaben des Frameworks

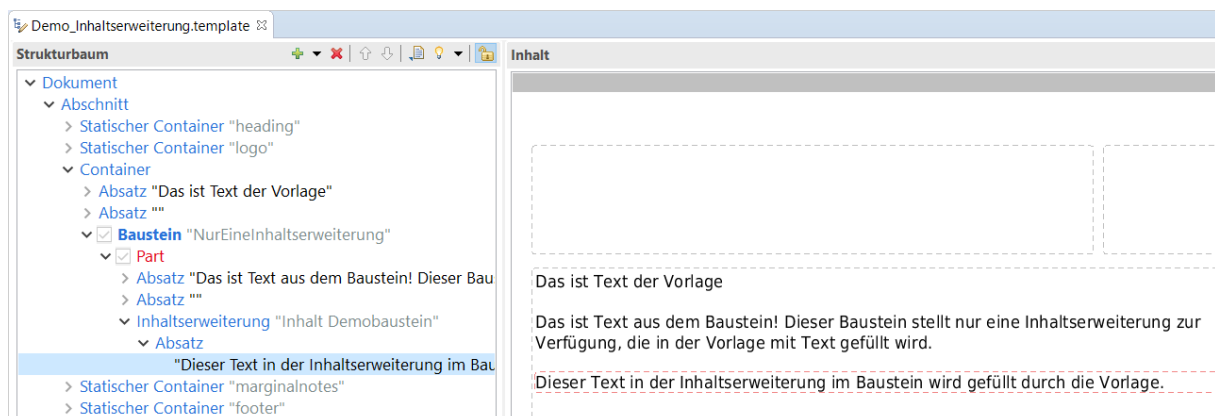
Das Framework kapselt die gesamte Logik der Auftragssteuerung, die zur Erzeugung der Korrespondenzen und Exemplare notwendig ist. Ein Vorlagenentwickler kann sich so auf die Erstellung seiner Vorlage konzentrieren und benötigt kein Wissen über diese Logik. Diese Auftragssteuerung ist weiter unten im Kapitel 3.3 Auftragssteuerung: Korrespondenzen und beschrieben.

Weiterhin stellt das Framework die Bausteine bereit, welche einen Teildokumenttypen mit den Standard-Inhalten füllen.

#### 3.1 Exkurs Inhaltserweiterungen

Das Framework nutzt bei allen Teildokumenten das TONIC-Konzept der Inhaltserweiterungen. Das bedeutet, dass ein Baustein eine Inhaltserweiterung zur Verfügung stellt und diese in der Vorlage mit Inhalt gefüllt wird. So bleibt der Framework-Baustein unangetastet.

Ein sehr einfaches Beispiel zur Nutzung einer Inhaltserweiterung finden Sie in Vorlage Demo\_Inhaltserweiterung.template. Dort wird der Baustein NurEineInhaltserweiterung.model eingebunden, welcher eigenen Text sowie die Inhaltserweiterung „Inhalt Demobaustein“ enthält. Die Inhaltserweiterung wird dann in der Vorlage mit Text gefüllt.



Dies ist auch in der Code-Ansicht der Vorlage erkennbar.

```
<RootPart xmlns="urn:kwsoft:mtext:tonic:dom">
  <DataDefinition></DataDefinition>
  <ModificationRights>
    <Allowed>
      <ModificationRight role="_EVERYONE_" operations="INPUT"></ModificationRight>
    </Allowed>
    <Denied></Denied>
  </ModificationRights>
  <Document>
    <Section>
      <Style parentName="Leaflet"></Style>
      <Container>
        <Par>
          <Span>
            <Text>Das ist Text der Vorlage</Text>
          </Span>
        </Par>
        <Par>
          <Span>
            <Text></Text>
          </Span>
        </Par>
        <ContainerPartRef uri="Models\NurEineInhaltserweiterung.model">
          <Extensions>
            <ContainerExtension id="Inhalt Demobaustein">
              <Par>
                <Span>
                  <Text>Dieser Text in der Inhaltserweiterung im Baustein wird gefüllt durch die Vorlage.</Text>
                </Span>
              </Par>
            </ContainerExtension>
          </Extensions>
        </ContainerPartRef>
      </Container>
    </Section>
  </Document>
</RootPart>
```

Inhaltserweiterungen können geschachtelt werden. Dazu wurde die Beispieldvorlage Demo\_Inhaltserweiterung\_geschachtelt.template bereitgestellt. Diese bindet den Baustein NurEineInhaltserweiterung\_1.model ein, welcher wiederum den Baustein NurEineInhaltserweiterung\_2.model einbindet. Beide Bausteine stellen ihre eigenen Inhaltserweiterungen zur Verfügung, die jedoch ineinander verschachtelt sind.

```
NurEineInhaltserweiterung_1.model
<ContainerPart xmlns="urn:kwsoft:mtext:tonic:dom">
  <DataDefinition></DataDefinition>
  <ContainerPartRef uri="Models\NurEineInhaltserweiterung_2.model">
    <Extensions>
      <ContainerExtension id="Inhalt Demobaustein2">
        <ContainerExtensionDef id="Inhalt Demobaustein1"></ContainerExtensionDef>
      </ContainerExtension>
    </Extensions>
  </ContainerPartRef>
</ContainerPart>
```

In der Vorlage kann nun die Inhaltserweiterung im 2. Baustein gefüllt werden.

## 3.2 Inhaltserweiterungen im Framework

Jede Vorlage muss den Baustein Korrespondenz einbinden. Die Inhaltserweiterung „Korrespondenz Inhalt“ enthält alle Teildokumente des Auftrags. Ihr kann z. B. als erstes der Baustein Anschreiben hinzugefügt werden. Für den Vorlagenentwickler ist dann dessen Inhaltserweiterung „Anschreiben Inhalt“ zu füllen.

Aus der Sichtweise des Vorlagenentwickler wird nur die durch ihn zu befüllende Inhaltserweiterung angezeigt.



Die folgende Grafik soll die Verschachtelung der Inhaltserweiterungen im Framework verdeutlichen. Der Vorlagenentwickler bindet zur Steuerung der Korrespondenzen und Exemplare zunächst den Message-Baustein ein. Der von dem Message-Baustein eingebundene InstanceControl-Baustein ist für den Vorlagenentwickler nicht zu sehen. Stattdessen werden die benötigten Teildokument-Bausteine, wie z. B. Letter oder TermsAndConditions, hinzugefügt und deren Inhaltserweiterung mit Text gefüllt.



## 3.3 Auftragssteuerung: Korrespondenzen und Exemplare

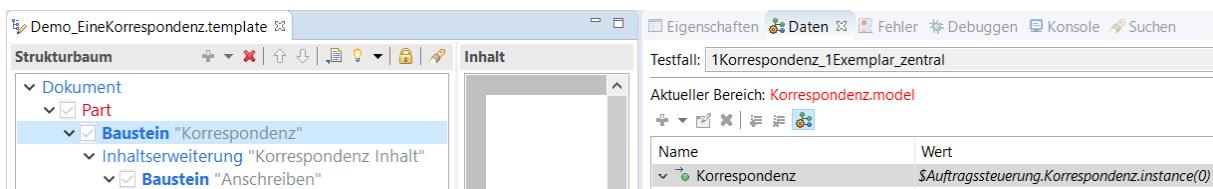
### 3.3.1 Korrespondenz-Baustein

Zur Auftragssteuerung, die zur Ausgabe aller Exemplare einer Korrespondenz führt, wird in einer Vorlage der Baustein Korrespondenz.model eingebunden.

Folgende Parameter werden von diesem benötigt:

Parameter	Beschreibung
Korrespondenz	Übergabe der aktuellen Korrespondenz
Betreff1, Betreff2, Betreff3	Bis zu drei Betreffzeilen

Wenn die Vorlage mehrere Korrespondenzen ausgibt, muss jedem Korrespondenz-Baustein die entsprechende Korrespondenz aus der Auftragssteuerung übergeben werden.

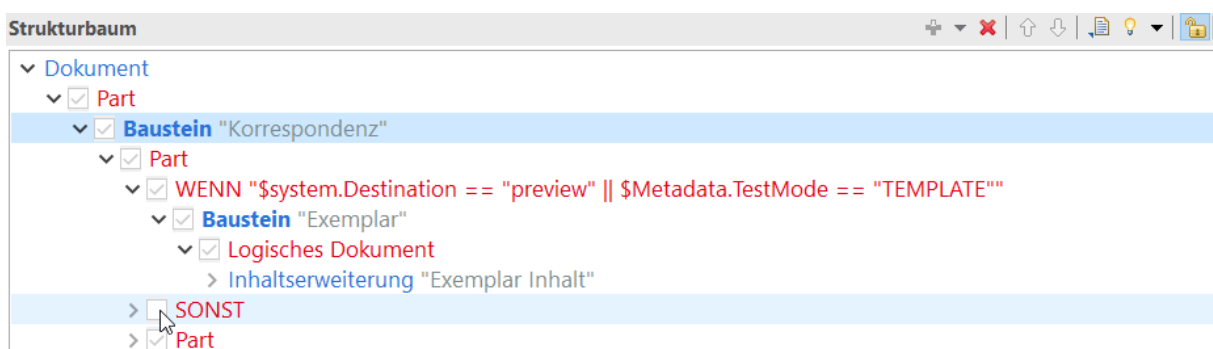


### 3.3.2 Alle Exemplare in der M/Workbench anzeigen

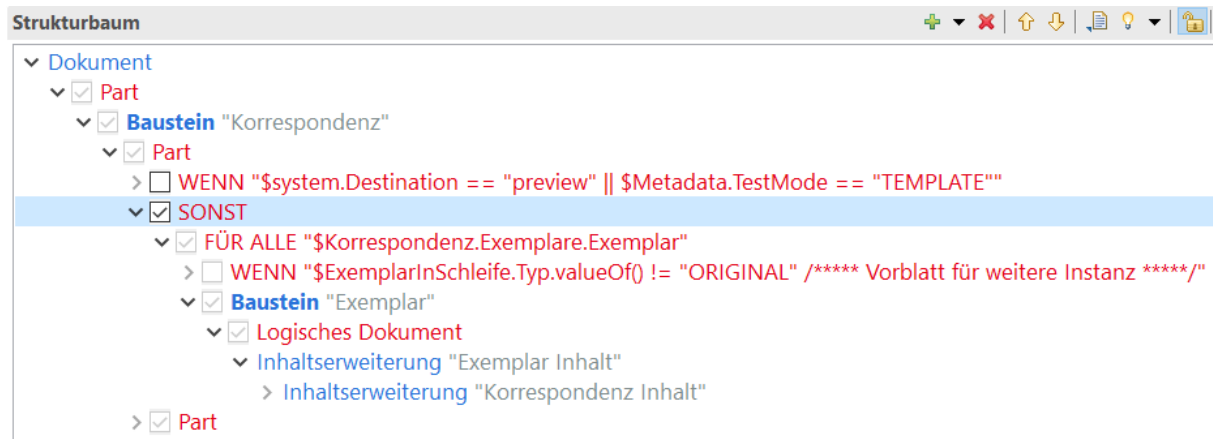
Standardmäßig wird in der M/Workbench und im M/TEXT TONIC Editor ausschließlich das Erst-Exemplar angezeigt. Die Ausgabe aller Exemplare erfolgt nur dann, wenn das Dokument finalisiert und an M/OMS übergeben wird.

Es gibt aber auch Anwendungsfälle, bei denen gewünscht ist, dass in der M/Workbench schon alle Exemplare angezeigt werden (z. B., um die Frameworksteuerung zu überprüfen).

Die Umschaltung ist mit einem Klick im Strukturbaum möglich.



Für den Fall, dass die Framework Bausteine ausgeblendet sind, klicken Sie einmal auf das Schloss Symbol oben rechts. Anschließend aktivieren sie die Checkbox des SONST-Zweiges im Baustein Korrespondenz.



Nach dem Umschalten wird durch das Framework eine Schleife über alle Exemplare durchlaufen.

<p>SAM AG - Hauptstr. 42 - 11073 Berlin</p> <p>Dr. Peter Friedrich Am Klinikum 31 47929 Doktorendorf</p>	 <p><b>SAM AG</b> Ihre Ansprechpartnerin <u>Assp</u> <u>ehardt</u> Hans Brackhaus Telefon: 0815-47111337 Fax: 0815-47114242</p>									
<p><b>Übernahme der Behandlungskosten</b> Patientin Maria Mercurial</p>										
<p>Sehr geehrter Herr Dr. Friedrich, für Ihre Patientin Frau Maria <u>Mercurial</u> werden wir die Kosten der Behandlung übernehmen.</p>										
<p>Mit freundlichen Grüßen</p>										
 <p>Brackhaus</p>										
<div data-bbox="205 1323 236 1525" style="border: 2px solid red; padding: 5px; width: fit-content;"> <p>%%KONSKV%%S01S(D)S%%</p> </div> <table border="1" data-bbox="261 1467 710 1507"> <tr> <td>Handelsregister: Amtsgericht Freiburg HRB 5674321</td> <td>Vorstand: Dr. Michaela <u>Bogedner</u> (Vorsitzender)</td> <td>Stammstift</td> </tr> <tr> <td>Sitz der Gesellschaft: Freiburg</td> <td>George <u>Pogatz</u> (2. Vorsitzender)</td> <td>Lernweg: 1-3</td> </tr> <tr> <td>Umsatzsteuer-Identifikationsnummer DE 567894321</td> <td>Franziska Becker, Dr. Thome <u>Schäfer</u></td> <td>71110 Freiburg</td> </tr> </table>		Handelsregister: Amtsgericht Freiburg HRB 5674321	Vorstand: Dr. Michaela <u>Bogedner</u> (Vorsitzender)	Stammstift	Sitz der Gesellschaft: Freiburg	George <u>Pogatz</u> (2. Vorsitzender)	Lernweg: 1-3	Umsatzsteuer-Identifikationsnummer DE 567894321	Franziska Becker, Dr. Thome <u>Schäfer</u>	71110 Freiburg
Handelsregister: Amtsgericht Freiburg HRB 5674321	Vorstand: Dr. Michaela <u>Bogedner</u> (Vorsitzender)	Stammstift								
Sitz der Gesellschaft: Freiburg	George <u>Pogatz</u> (2. Vorsitzender)	Lernweg: 1-3								
Umsatzsteuer-Identifikationsnummer DE 567894321	Franziska Becker, Dr. Thome <u>Schäfer</u>	71110 Freiburg								

Darüber hinaus werden Platzhalter für Steuerzeichen am Blattrand ausgegeben.

## 4 Beispiel-Teildokumenttypen

Teildokumenttypen definieren das Aussehen einer oder mehrere Seiten des Dokuments. Die Beispiele beschreiben die im Framework vordefinierten Teildokumenttypen. Dies sind zurzeit:

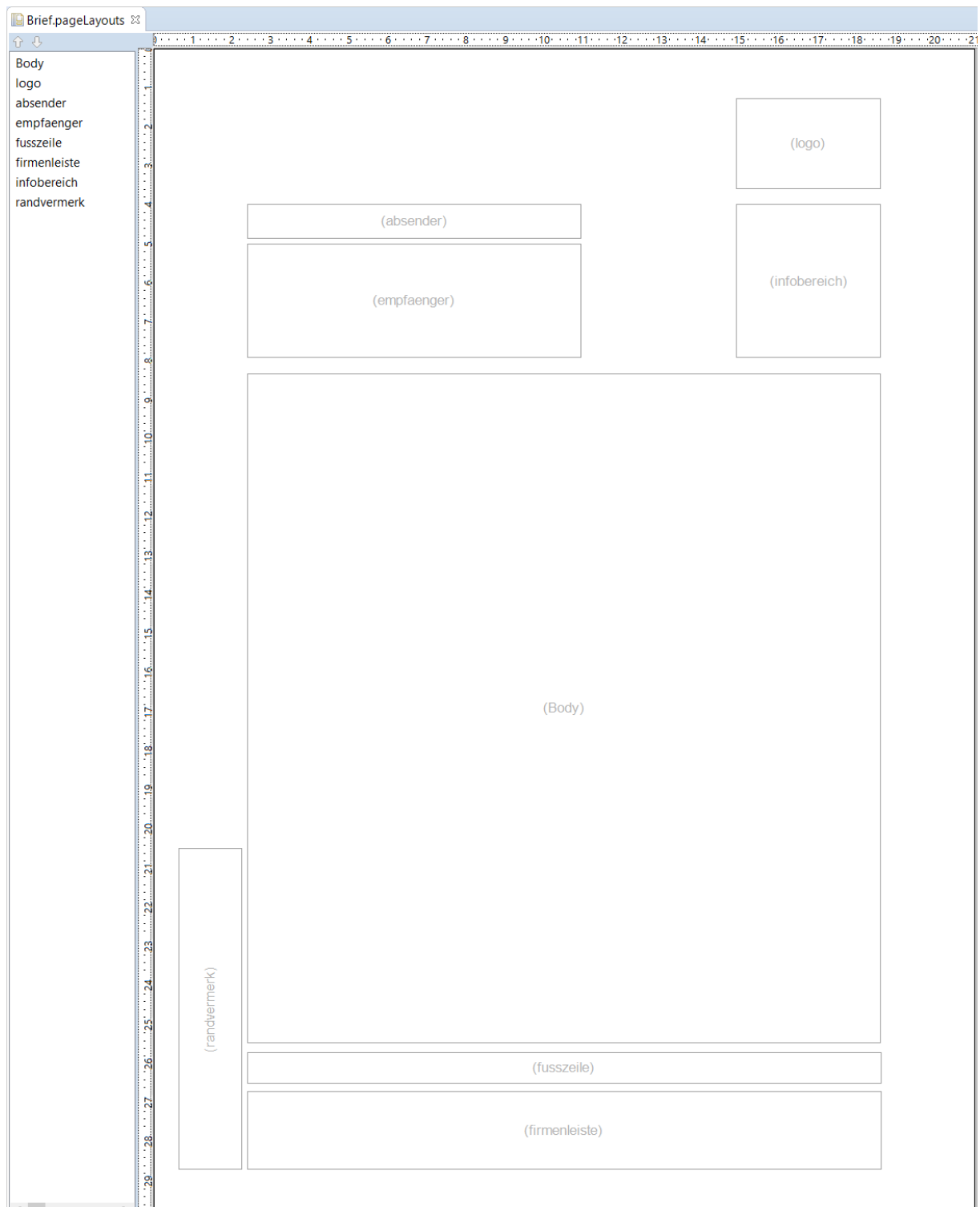
- Brief
- AGB
- Anlage
- Infoblatt
- Rückantwort

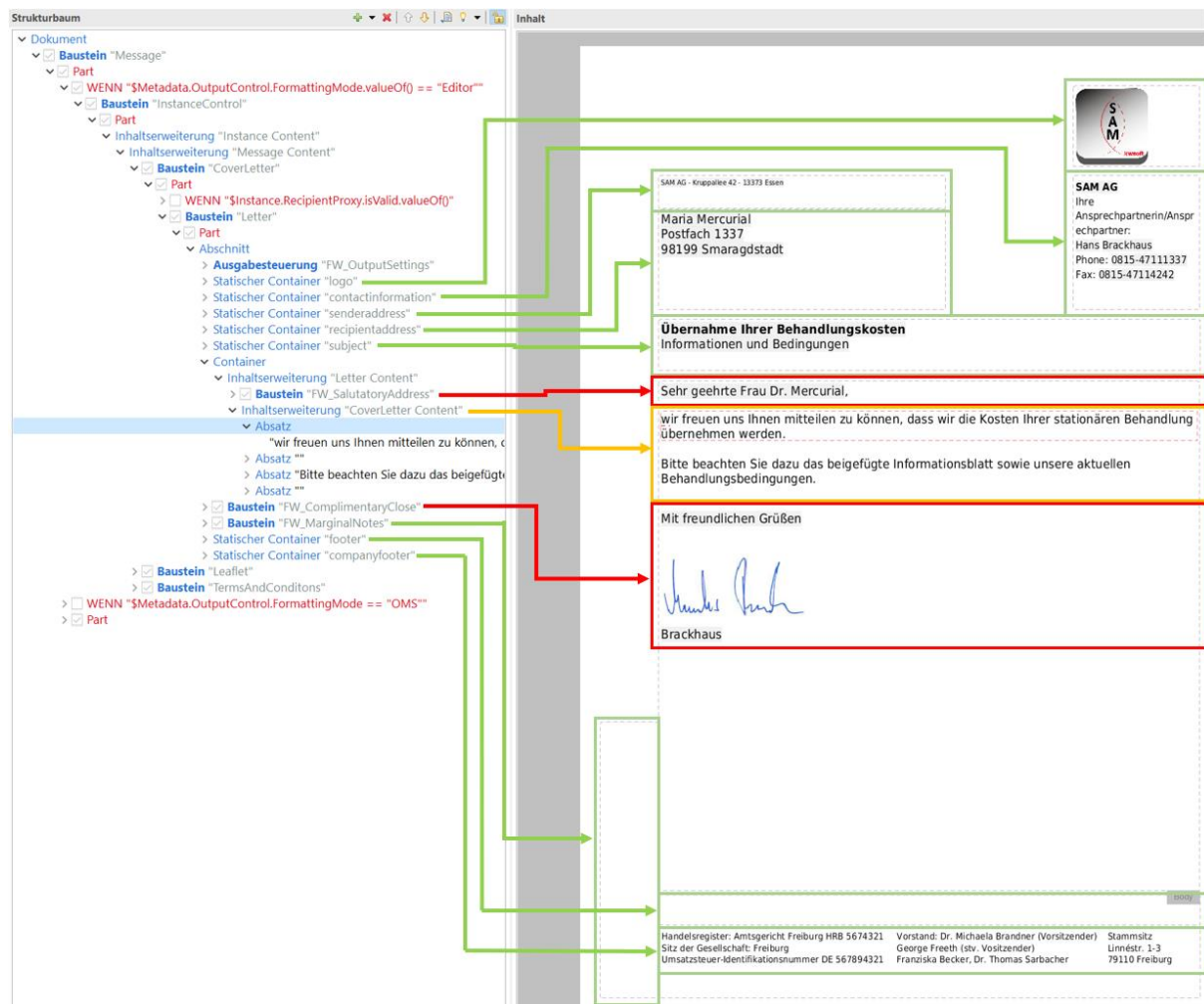
## 4.1 Allgemeine Layoutelemente

Teildokumente bestehen aus verschiedenen Bausteinen, die einzelne Textausgaben durchführen. Allgemeine Bausteine, die in mehreren Teildokumenten benutzt werden, sind im Ordner Framework\Layoutelemente zu finden. Bausteine, die spezifisch für ein Teildokument sind, liegen im jeweiligen Unterordner des Teildokuments. Beispielsweise ist der Baustein für die Briefanrede, **FW\_Briefanrede.model** im Teildokument Brief, zu finden im Ordner Framework\Teildokumente\Brief.

## 4.2 Brief-Teildokument

Das Brief-Teildokument stellt alle Bereiche (Regions) bereit, die für ein Anschreiben notwendig sind. Der eigentliche Briefinhalt wird über die Inhaltserweiterung „Brief Inhalt“ gefüllt. Somit ist sichergestellt, dass nur die Vorlage den Dokumenteninhalt bereitstellt und die Frameworkbausteine unangetastet bleiben.



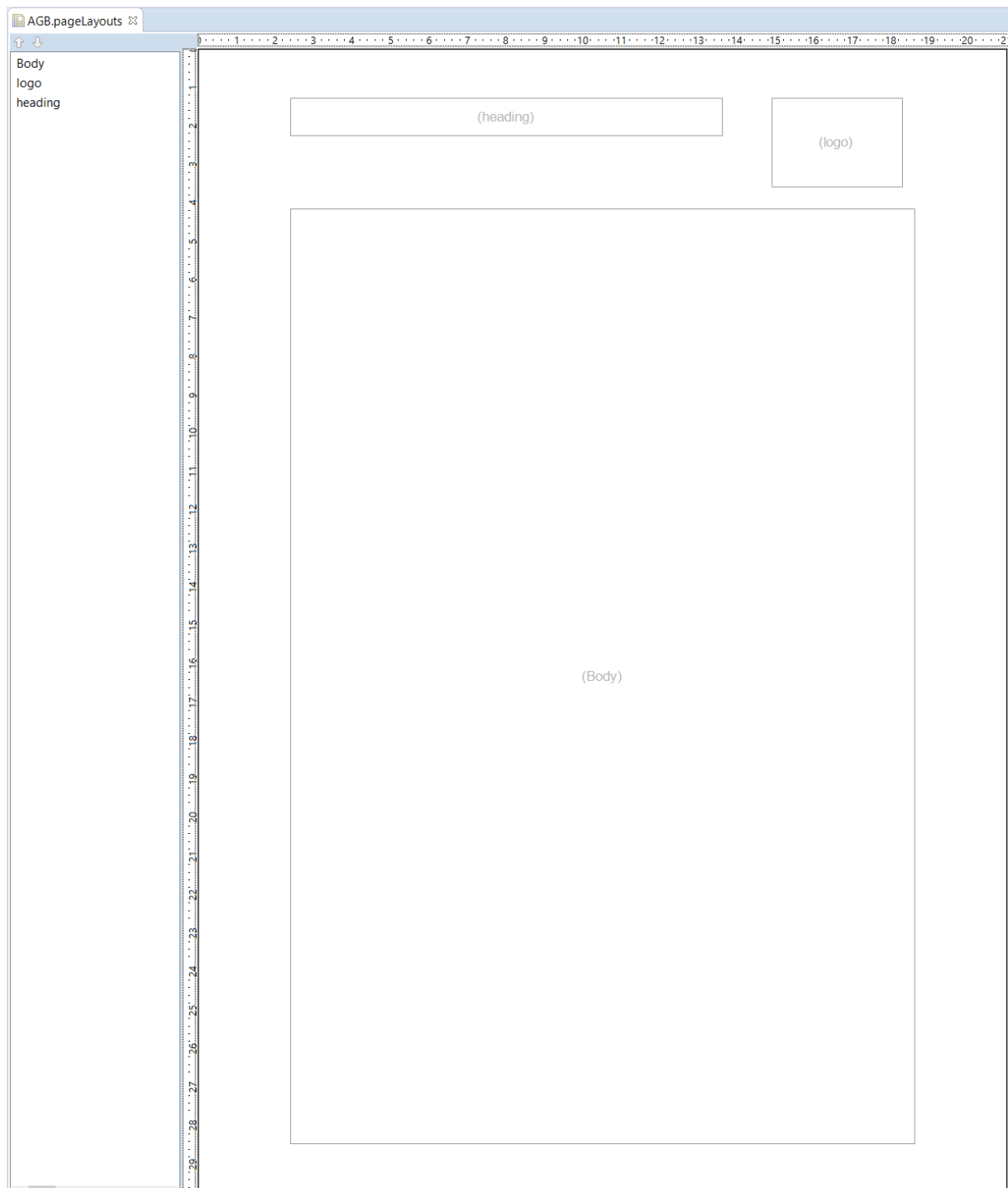


In diesem Diagramm sind die Frameworkbausteine in statischen Regionen **grün** dargestellt. Frameworkbausteine in der Body-Region sind **rot**, der eigentliche Dokumentinhalt ist **orange** hervorgehoben.

#### 4.2.1 Frameworkbausteine im Brief-Teildokument

Baustein	Beschreibung	Parameter
FW_Logo.model	Der Bereich rechts oben stellt ein Logo bereit	keine
FW_Absender.model	Brief-Absender oberhalb der Anschrift	Absender.datamodel
FW_Empfaenger.model	Brief-Anschrift	Empfaenger.datamodel
FW_Betreff.model	Ein bis drei Betreffzeilen	Betreff1, Betreff2, Betreff3
FW_Briefanrede.model	Anrede-Zeile	Empfaenger.datamodel
FW_Gruss.model	Grusszeile	Absender.datamodel
FW_Firmenleiste.model	Handelsrechtliche Angaben	keine







## 4.4 Anlage-Teildokument

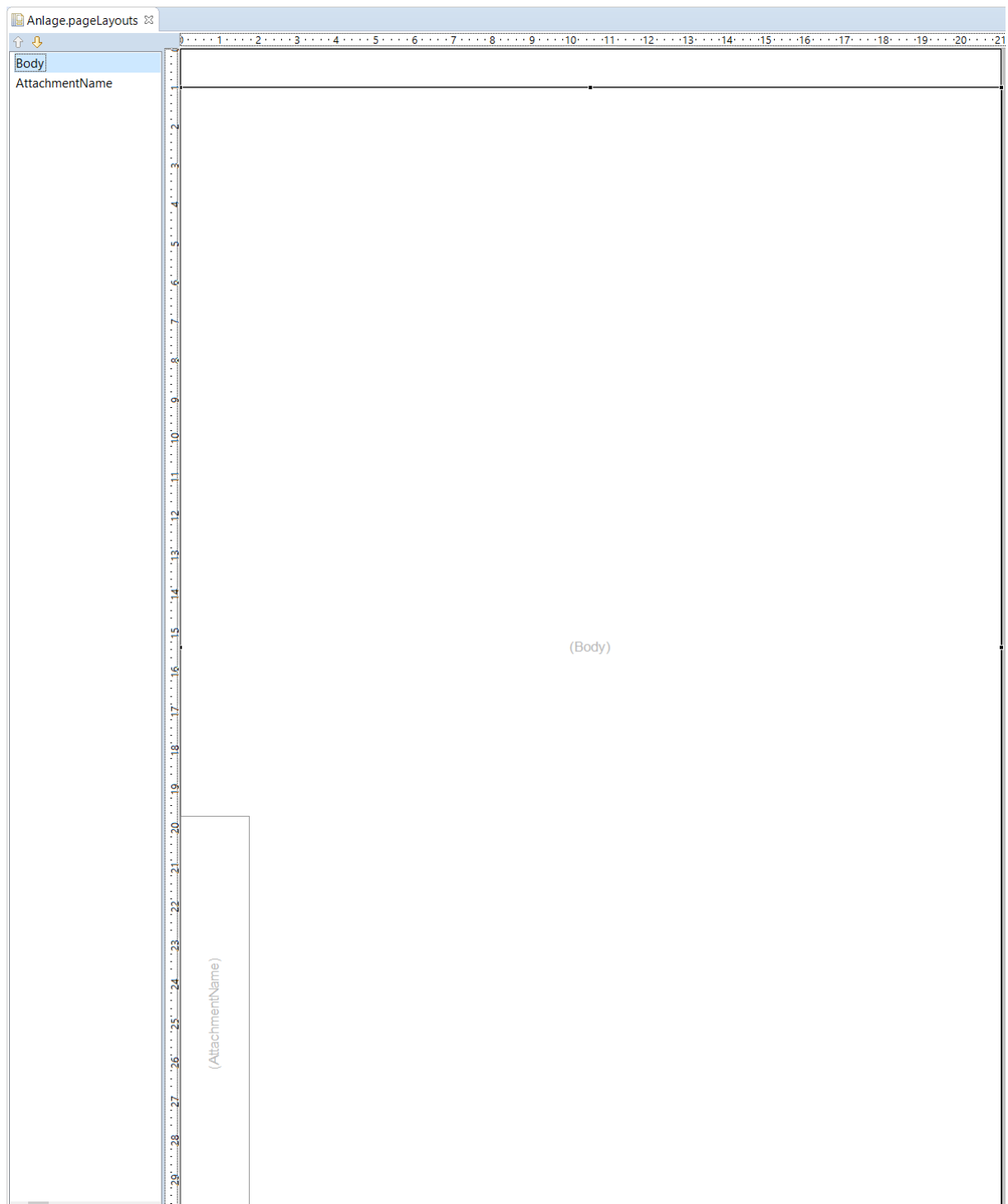
Das Anlage-Teildokument zeigt, wie man externe Ressourcen, wie z. B. ein PDF, einbindet. Als Parameter wird das PDF als Anlage sowie der Anlage-Namen übergeben. Das Anlage-Teildokument besitzt nur eine Body-Region und die Region AttachmentName.

The screenshot displays the Tonic framework interface. On the left, the 'Strukturbaum' (Structure Tree) shows a document structure with the following elements:

- Dokument
  - Part
    - Baustein "Korrespondenz"
      - Inhaltserweiterung "Korrespondenz Inhalt"
        - Baustein "Anschieben"
          - Baustein "Anlage" (highlighted)

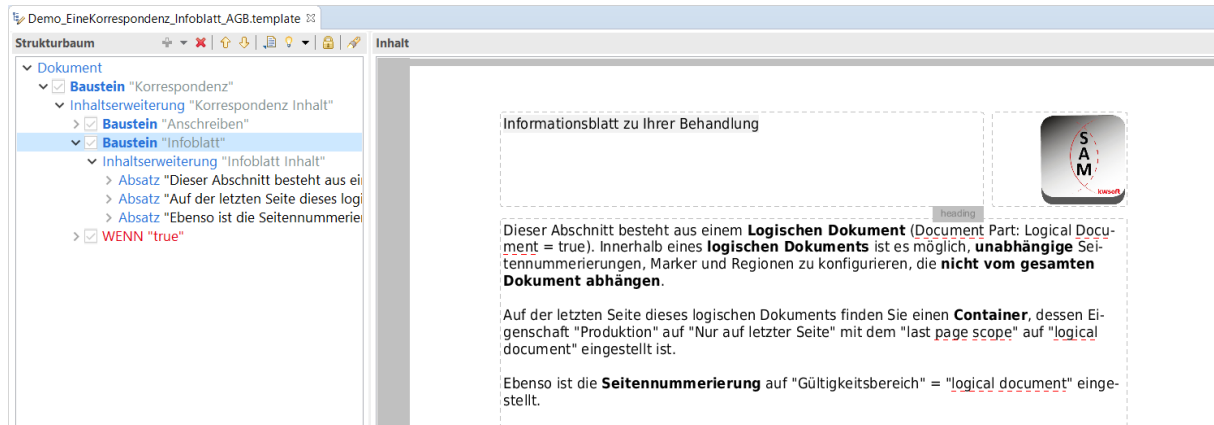
On the right, the 'Inhalt' (Content) area shows a rendered document preview. The preview includes:

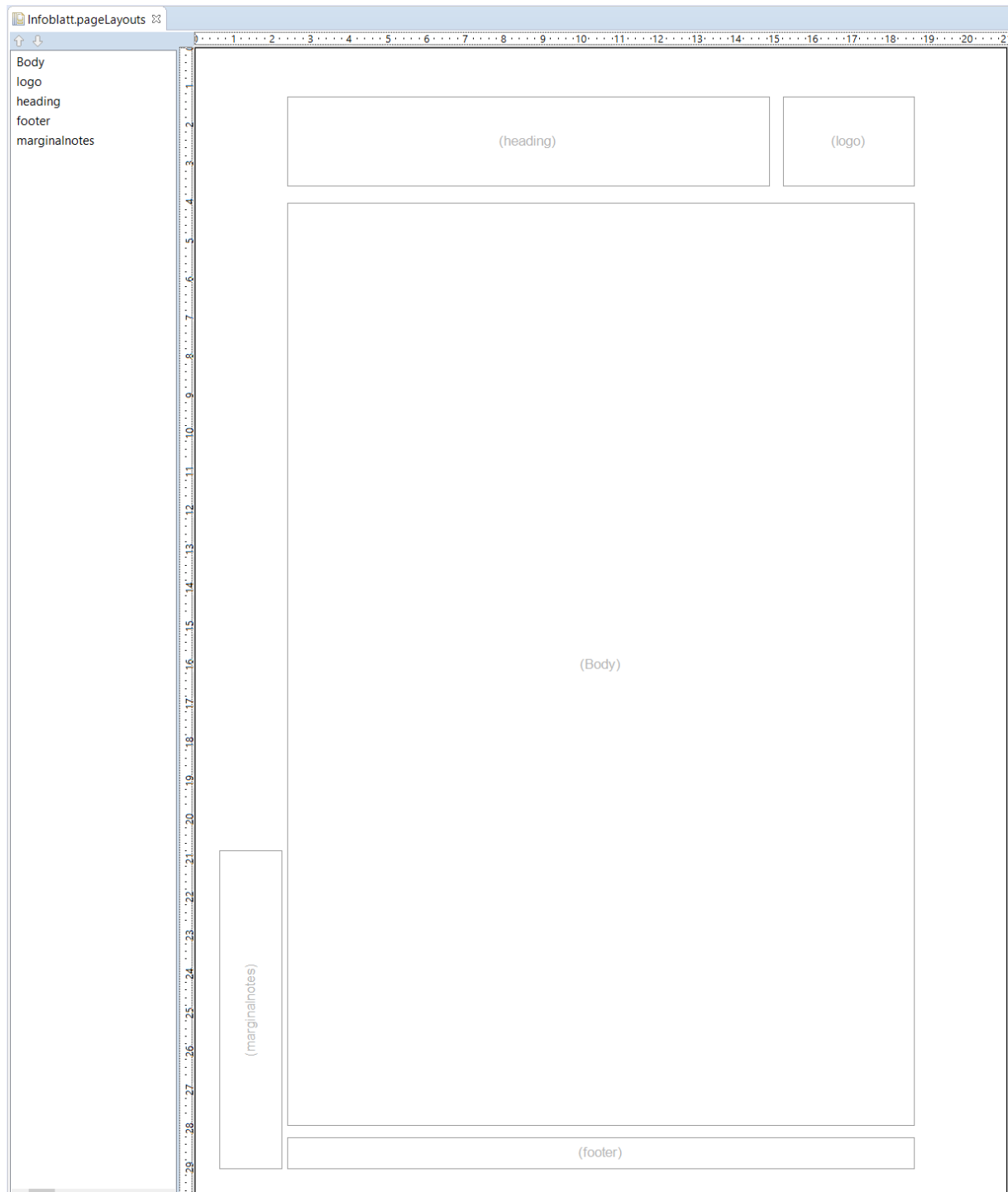
- A header section with the company logo (S A M) and name 'Aktiengesellschaft'.
- Contact information: Alfred-Döblin-Platz 59, 79100 Freiburg, Tel. +49 761 45689-0, Fax +49 761 45689-1115, Mail service@sam.net.
- A section titled 'SEPA-Basislastschrift (Wiederkehrende Zahlungen)'.
- A paragraph of text: 'Ich ermächtige die sam Aktiengesellschaft · Alfred-Döblin-Platz 59 · 79100 Freiburg Zahlungen von meinem Konto mittels Lastschrift einzuziehen. Zugleich weise ich mein Kreditinstitut an, die von der sam AG auf mein Konto gezogenen Lastschriften einzulösen.'
- A note: 'Hinweis: Ich kann innerhalb von acht Wochen, beginnend mit dem Belastungsdatum, die Erstattung des belasteten Betrages verlangen.'



## 4.5 Infoblatt-Teildokument

Das Infoblatt-Teildokument stellt ein einfaches Layout zur Verfügung. Folgende Parameter werden übergeben: Titel, DoNotArchive, SendAsAttachment, AttachmentName. Der Inhalt wird in der Vorlage über die Inhaltserweiterung „Infoblatt Inhalt“ gefüllt.






## 4.6 Rückantwort-Teildokument

Das Rückantwort-Teildokument dient dazu, dem Empfänger die Möglichkeit zu geben, auf das Schreiben des Absenders mittels eines vorgegebenen Dokuments zu antworten. Dies kann zum Beispiel ein Formular sein, welches der Empfänger ausfüllen und zurücksenden soll.

Ein Beispiel ist in Vorlage Demo\_Formular.template zu finden, die ein ausfüllbares Formular erzeugt.

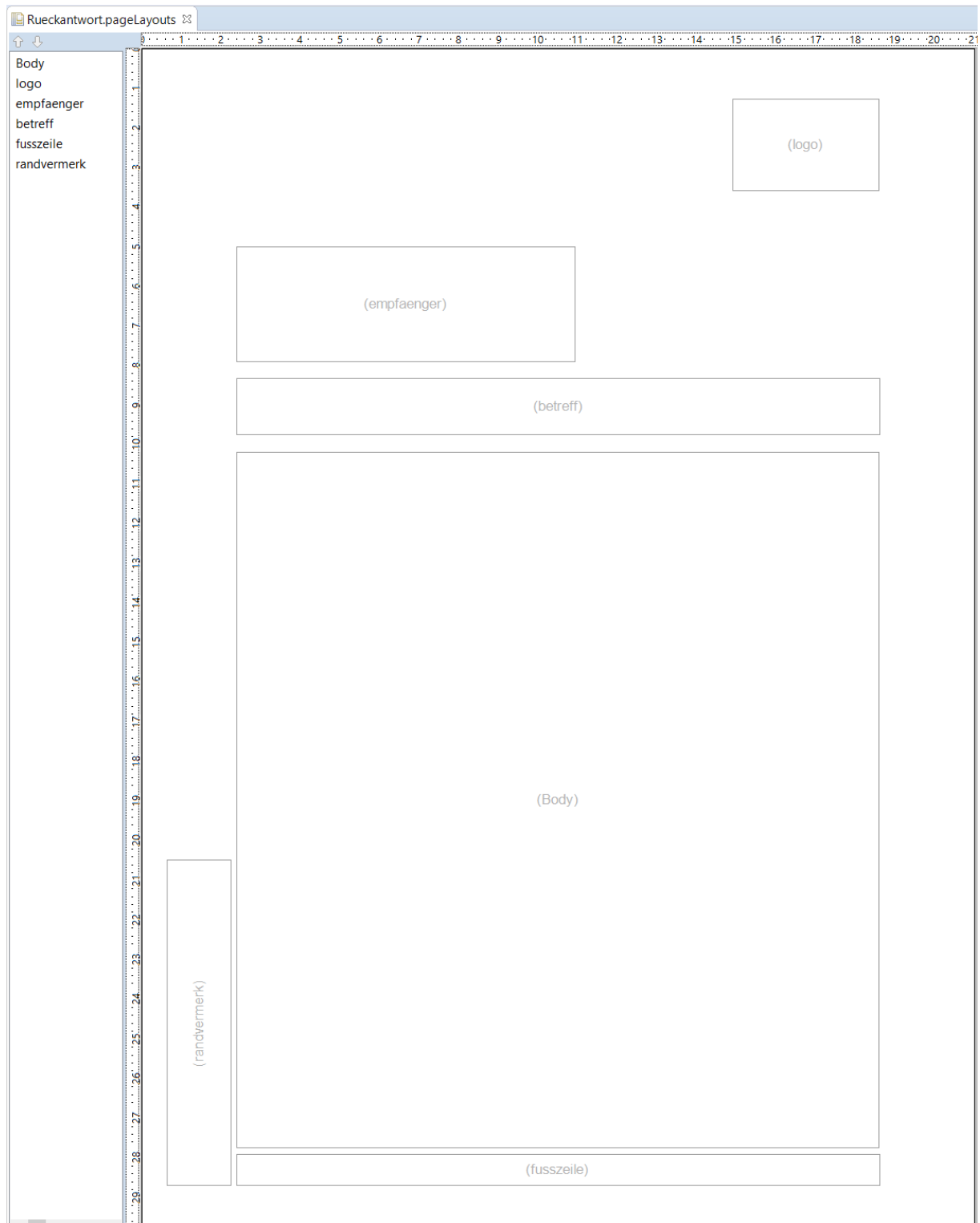


k&w Krankenversicherung  
Linnéstr. 1-3  
79110 Freiburg

**Antrag auf Haushaltshilfe**

Die Erhebung der persönlichen Daten (Sozialdaten) ist aufgrund der §§284 und 206 Sozialgesetzbuch (SGB) Fünftes Buch (V) in Verbindung mit §21 SGB Zehntes Buch (X) erforderlich.

Name, Vorname	Geburtsdatum
<input style="width: 95%;" type="text"/>	<input style="width: 95%;" type="text"/>
Für Rückfragen tagsüber telefonisch erreichbar (Angabe ist freiwillig)	Familienstand
<input style="width: 95%;" type="text"/>	<input style="width: 95%;" type="text"/>
Der Betrag soll auf folgendes Konto überwiesen werden:	
BIC	IBAN
<input style="width: 95%;" type="text"/>	<input style="width: 95%;" type="text"/>
Geldinstitut	
<input style="width: 95%;" type="text"/>	
Kontoinhaber (falls abweichend)	
<input style="width: 95%;" type="text"/>	
Mir ist lt. ärztlicher Bescheinigung die Weiterführung des Haushaltes nicht möglich wegen...	
<input type="radio"/> Schwangerschaft / Entbindung <input type="radio"/> einer akuten Erkrankung	
Die Erkrankung ist die Folge	
<input type="radio"/> eines Arbeitsunfalls/einer Berufskrankheit <input type="radio"/> eines privaten Unfalls	<input type="radio"/> ja <input type="radio"/> nein
	<input type="radio"/> ja <input type="radio"/> nein
	Unfalltag
	<input style="width: 95%;" type="text"/>
Datum, Unterschrift der/des Versicherten	



## 5 Styles

Alles verfügbaren Stile werden in der **Default.styleCollection**-Datei definiert. Dazu gehören u.a. die Seitenlayouts und die Absatzschriften.

### Seitenlayouts

```
<SectionStyle name="Brief" title="Brief" title-en_US="Letter" title-de_DE="Brief">
...
  <ResponsiveLayoutRef
uri="Framework\Teildokumente\Brief\Brief.responsiveLayout"></ResponsiveLayoutRef>
  <PageLayouts parentRef="Framework/Teildokumente/Brief/Brief.pageLayouts"></PageLayouts>
</SectionStyle>

<SectionStyle name="Infoblatt">
  <ResponsiveLayoutRef
uri="Framework\Teildokumente\Infoblatt\Infoblatt.responsiveLayout"></ResponsiveLayoutRef>
  <PageLayouts
parentRef="Framework/Teildokumente/Infoblatt/Infoblatt.pageLayouts"></PageLayouts>
</SectionStyle>

<SectionStyle name="AGB">
  <ResponsiveLayoutRef
uri="Framework\Teildokumente\AGB\AGB.responsiveLayout"></ResponsiveLayoutRef>
  <PageLayouts parentRef="Framework/Teildokumente/AGB/AGB.pageLayouts"></PageLayouts>
</SectionStyle>

<SectionStyle name="Rueckantwort">
...
  <PageLayouts
parentRef="Framework/Teildokumente/Rueckantwort/Rueckantwort.pageLayouts"></PageLayouts>
</SectionStyle>

<SectionStyle name="Blank">
  <PageLayouts parentRef="Framework/Teildokumente/Blank/Blank.pageLayouts"></PageLayouts>
</SectionStyle>
```

### Seitenlayout für Anlagen

```
<AttachmentSectionStyle name="AnLage">
  <PageLayouts parentRef="Framework/Teildokumente/AnLage/AnLage.pageLayouts"></PageLayouts>
</AttachmentSectionStyle>
```

### Definition eines Absatz-Stils, wenn kein Stil angegeben wird

```
<ParStyle>
  <SpanStyle>
    <FontSize>11000.0mpt</FontSize>
    <FontFamily>STANDARD</FontFamily>
  </SpanStyle>
  <RightIndent>0.0cm</RightIndent>
  <Align>left</Align>
  <LeftIndent>0.0cm</LeftIndent>
</ParStyle>
```

Im Framework genutzte Stile beginnen grundsätzlich mit FW\_ und sind im TONIC-Editor verborgen (hideInEditor="true"). Diese sollten nicht für die Vorlagenentwicklung genutzt werden, da sie sich ändern können.

Beispiel für den Stil FW\_Default:

```
<ParStyle name="FW_Default" hideInEditor="true">
  <SpanStyle>
    <FontSize>11000.0mpt</FontSize>
    <FontFamily>STANDARD</FontFamily>
  </SpanStyle>
  <LineSpacing type="fixed" value="12.5pt"></LineSpacing>
  <RightIndent>0.0mpt</RightIndent>
  <Align>left</Align>
  <LeftIndent>0.0mpt</LeftIndent>
</ParStyle>
```

Stile, die für die Vorlagenerstellung genutzt werden, können mit beliebigen Namen versehen werden.

## 6 Beispielvorlagen

Vorlage	(Kapitel) Beschreibung
Demo_EineKorrespondenz.template	(1.2.9.1) Testfall 1Korrespondenz_1Exemplar_zentral erzeugt eine Korrespondenz mit einem Exemplar.  (1.3.3.1) Testfall 1Korrespondenz_1Exemplar_AlternativEmpfaenger erzeugt eine Korrespondenz mit einem Exemplar welches an einen alternativen Empfänger verschickt wird.
Demo_EineKorrespondenz_ZweiExemplare.template	(1.2.9.2) Testfall 1Korrespondenz_2Exemplare_zentral erzeugt eine Korrespondenz mit zwei Exemplaren. Das weitere Exemplar wird einem anderen Empfänger zugestellt.
Demo_ZweiKorrespondenzen.template	(1.2.9.3) Testfall 2Korrespondenzen_1Exemplar_zentral erzeugt zwei Korrespondenzen mit jeweils einem Exemplar. Die Korrespondenzen haben dabei einen unterschiedlichen Inhalt.
Demo_EineKorrespondenz_Infoblatt_AGB.template	(1.4.1) Testfall 1Korrespondenz_1Exemplar erzeugt eine Korrespondenz mit einem Exemplar, das aus verschiedenen Teildokumenten besteht.
Demo_EineKorrespondenz_ZweiExemplare_Infoblatt.template	(1.4.2.1) Testfall 1Korrespondenz_2Exemplare_zentral erzeugt eine Korrespondenz mit zwei Exemplaren. Das Erst-Exemplar enthält ein Infoblatt. Das weitere Exemplar enthält dieses Infoblatt nicht.
Demo_Inhaltserweiterung.template Demo_Inhaltserweiterung_geschachtelt.template	(3.1) Diese beiden Vorlagen zeigen die Nutzung von Inhaltserweiterungen.
Demo_Formular.template	(4.6) Diese Vorlage erzeugt mit Hilfe des Rückantwort-Teildokuments ein ausfüllbares Formular.



## 7 M/OMS Ansteuerung

### 7.1 Baustein Ausgabesteuerung

Jeder Teildokument-Baustein (wie z. B. Brief.model) bindet die Ausgabesteuerung FW\_Ausgabesteuerung.outputSettings ein. Diese regelt, welche Datenmodellknoten aus Auftragssteuerung.datamodel auf die M/OMS-Parameter aus der moms.conf.xml gemapped werden.

#### 7.1.1 ZENTRALDRUCK

Zunächst wird über alle Mehrfachknoten **Ausgabesteuerung\*** des Exemplars iteriert, um den richtigen Knoten zu finden:

▼ Logik	
Schleifenelement-Alias	\$Ausgabesteuerung
Schleife über	\$ExemplarAusgabesteuerung.Ausgabesteuerung
Sichtbar wenn	\$Ausgabesteuerung.Ausgabekanal.valueOf() == 'ZENTRALDRUCK'

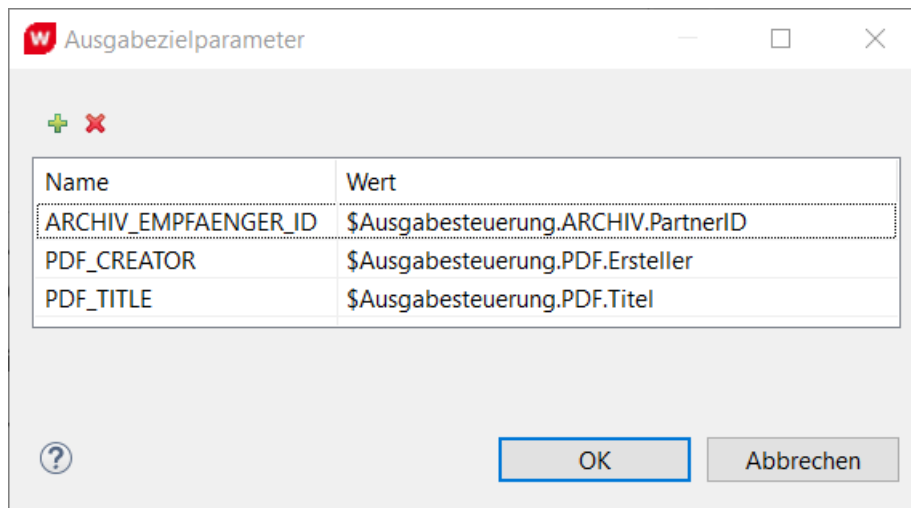
Ist dieser gefunden, werden die OMS-Parameter gesetzt:

Name	Wert
ZENTRALDRUCK_BATCH	\$Ausgabesteuerung.ZENTRALDRUCK.Batch
EMPFAENGER_PLZ	\$Ausgabesteuerung.ZENTRALDRUCK.PLZ
EMPFAENGER_LKZ	\$Ausgabesteuerung.ZENTRALDRUCK.LKZ
EMPFAENGER_ID	\$Ausgabesteuerung.ZENTRALDRUCK.PartnerID

#### 7.1.2 ARCHIV

Hier wird zusätzlich geprüft, ob der Parameter \$NichtArchivieren gesetzt ist. Ist das der Fall, wird das Teildokument nicht archiviert.

▼ Logik	
Schleifenelement-Alias	\$Ausgabesteuerung
Schleife über	\$ExemplarAusgabesteuerung.Ausgabesteuerung
Sichtbar wenn	\$Ausgabesteuerung.Ausgabekanal.valueOf() == 'ARCHIV' && !\$NichtArchivieren.valueOf()

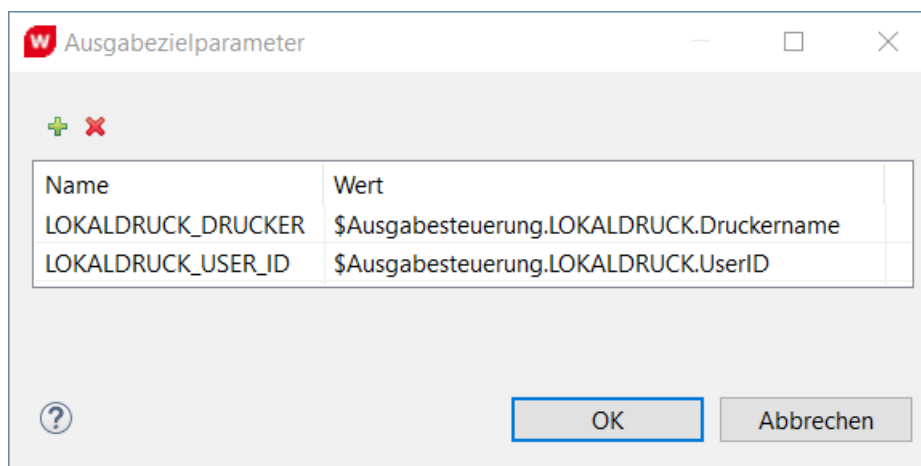


Name	Wert
ARCHIV_EMPFAENGER_ID	\$Ausgabesteuerung.ARCHIV.PartnerID
PDF_CREATOR	\$Ausgabesteuerung.PDF.Ersteller
PDF_TITLE	\$Ausgabesteuerung.PDF.Titel

Buttons: OK, Abbrechen

### 7.1.3 LOKALDRUCK

Siehe ZENTRALDRUCK, nur mit folgenden OMS-Parametern.

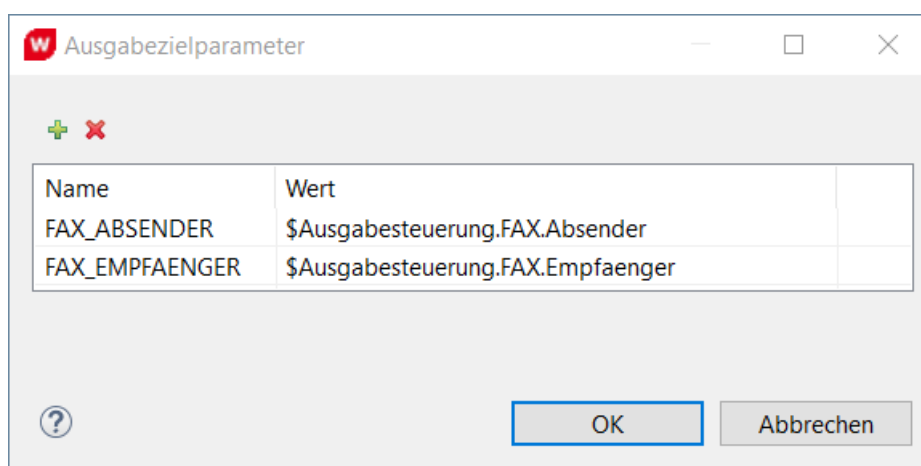


Name	Wert
LOKALDRUCK_DRUCKER	\$Ausgabesteuerung.LOKALDRUCK.Druckername
LOKALDRUCK_USER_ID	\$Ausgabesteuerung.LOKALDRUCK.UserID

Buttons: OK, Abbrechen

### 7.1.4 FAX

Siehe ZENTRALDRUCK, nur mit folgenden OMS-Parametern.



Name	Wert
FAX_ABSENDER	\$Ausgabesteuerung.FAX.Absender
FAX_EMPFAENGER	\$Ausgabesteuerung.FAX.Empfaenger

Buttons: OK, Abbrechen

### 7.1.5 EMAIL oder EMAILATTACHMENT

Hier wird eine Unterscheidung getroffen, ob das Teildokument als E-Mail-Text oder als E-Mail-Anhang versendet werden soll. Abhängig davon wird das Medium EMAIL bzw. EMAILATTACHMENT gewählt.

▼ Logik	
Schleifenelement-Alias	\$OutputSettings
Schleife über	\$InstanceOutputSettings.OutputSettings
Sichtbar wenn	\$OutputSettings.OutputChannel.valueOf() == 'EMAIL' && ! \$SendAsAttachment.valueOf()

Ausgabezielparameter

+

×

Name	Wert
MAIL_FROM	\$Ausgabesteuerung.EMAIL.Absender
MAIL_TO	\$Ausgabesteuerung.EMAIL.Empfaenger
MAIL_SUBJECT	\$Ausgabesteuerung.EMAIL.Betreff.isEmpty() ? \$Betreff1.valueOf() : \$Ausgabesteuerung.EMAIL.Betreff.valueOf()

?

OK

Abbrechen

Ausgabezielparameter

+

×

Name	Wert
MAIL_FROM	\$Ausgabesteuerung.EMAIL.Absender
MAIL_TO	\$Ausgabesteuerung.EMAIL.Empfaenger
MAIL_SUBJECT	\$Ausgabesteuerung.EMAIL.Betreff
MAIL_ATTACHMENTNAME	\$AttachmentName

?






















OK

Abbrechen

## 7.2 M/OMS Konfiguration

Die Konfiguration des M/OMS besteht aus Standard-Funktionen und wird deshalb hier nur kurz erläutert.

In den Parametern werden die aus der Vorlage gefüllten Parameter definiert.

- ✓  Parameter
  -  ARCHIV\_DATA
  -  ARCHIV\_EMPFAENGER\_ID
  -  EMPFAENGER\_ID
  -  EMPFAENGER\_LKZ
  -  EMPFAENGER\_NAME
  -  EMPFAENGER\_ORT
  -  EMPFAENGER\_PLZ
  -  FAX\_ABSENDER
  -  FAX\_EMPFAENGER
  -  LOKALDRUCK\_DRUCKER
  -  LOKALDRUCK\_USER\_ID
  -  MAIL\_ATTACHMENTNAME
  -  MAIL\_FROM
  -  MAIL\_SUBJECT
  -  MAIL\_TO
  -  PDF\_CREATOR
  -  PDF\_TITLE
  -  PRINTER\_NAME
  -  VORLAGE
  -  ZENTRALDRUCK\_BATCH












### 7.2.1 ZENTRALDRUCK-Stapel

Dieser Stapel legt eine AFP-Druckdatei in einem Verzeichnis ab. Zur Demonstration erzeugt er auf dem Ausdruck einen QR-Code. Außerdem werden je Dokument AFP-TLE-Werte integriert (siehe auch `mtext.conf.xml`).

- ✓ ZENTRALDRUCK
  - ✓ Dokument-Auswahl
    - KW\_MEDIUM = ZENTRALDRUCK
    - AND KW\_STATUS = WAIT\_FOR\_STACK\_PROCESSING
  - ✓ Stapel-Extensions
    - ✓ Parameter-Setter-Extension set file path and file name
      - > Setze Parameter - ZENTRALDRUCK\_PFAD
      - > Setze Parameter - ZENTRALDRUCK\_DATEINAME
    - 1111 Group-Values-Saver-Extension
    - ✓ Marking-Extension Datamatrix
      - ✓ Markierungen
        - > QR Code
    - ✓ Renderer-Extension render AFP
      - ✓ Metadaten für den Renderer
        - ✓ Metadaten für jedes Dokument oder jede Seitengruppe
          - Setze Schlüssel-Parameter LKZ = EMPFAENGER\_LKZ
          - Setze Schlüssel-Parameter PLZ = EMPFAENGER\_PLZ
    - ✓ Collect-Extension deposit AFP
      - Text-Ersetzungs-Parameter - PRINTFILE\_DOCUMENT\_COUNT = PRINTFILE\_DOCUMENT\_COUNT
      - Text-Ersetzungs-Parameter - PRINTFILE\_SHEET\_COUNT = PRINTFILE\_SHEET\_COUNT
      - Text-Ersetzungs-Parameter - RUN\_ID = KW\_RUN\_ID
      - Text-Ersetzungs-Parameter - STACK\_NAME = STACK\_NAME
      - Text-Ersetzungs-Parameter - STACK\_DSC = STACK\_DESCRIPTION
      - Text-Ersetzungs-Parameter - START\_TIME\_DATE = START\_TIME\_DATE
      - Text-Ersetzungs-Parameter - LKZ = EMPFAENGER\_LKZ
  - ✓ Dokument-Gruppen
    - CountryCode
    - Envelope
  - ✓ Sortierkriterien
    - EMPFAENGER\_LKZ aufsteigend
    - EMPFAENGER.ID aufsteigend



















## 7.2.2 ZENTRALDRUCK\_PDFVT-Stapel

Dieser Stapel demonstriert die Erzeugung einer Druckdatei im Format PDF/VT.

- ▼  ZENTRALDRUCK\_PDFVT
  - ▼  Dokument-Auswahl
    - KW\_MEDIUM = ZENTRALDRUCK
    - AND KW\_STATUS = WAIT\_FOR\_STACK\_PROCESSING
  - ▼  Sortierkriterien
    - EMPFAENGER\_PLZ aufsteigend
    - EMPFAENGER\_ID aufsteigend
  - ▼  Stapel-Extensions
    - ▼  Parameter-Setter-Extension Pfad und Dateiname setzen
      - >  Setze Parameter - ZENTRALDRUCK\_PFAD
      - >  Setze Parameter - ZENTRALDRUCK\_DATEINAME
    - ▼  Renderer-Extension PDF\_Renderer
      - Text-Ersetzungs-Parameter - PRINTFILE\_DOCUMENT\_COUNT = PRINTFILE\_DOCUMENT\_COUNT
      - Text-Ersetzungs-Parameter - PRINTFILE\_SHEET\_COUNT = PRINTFILE\_SHEET\_COUNT
      - Text-Ersetzungs-Parameter - RUN\_ID = KW\_RUN\_ID
      - Text-Ersetzungs-Parameter - STACK\_NAME = STACK\_NAME
      - Text-Ersetzungs-Parameter - STACK\_DSC = STACK\_DESCRIPTION
      - Text-Ersetzungs-Parameter - START\_TIME\_DATE = START\_TIME\_DATE
      - Text-Ersetzungs-Parameter - LKZ = EMPFAENGER\_LKZ
    -  Collect-Extension Collection PDF
  - ▼  Dokument-Gruppen
    -  Customer























### 7.2.3 ARCHIV-Stapel

Mit diesem Stapel wird eine Archivdatei im PDF-Format mit dazugehöriger Index-Datei im XML-Format erzeugt.

- ▼  ARCHIV
  - ▼  Dokument-Auswahl
    - KW\_MEDIUM = ARCHIV
    - AND KW\_STATUS = WAIT\_FOR\_STACK\_PROCESSING
  - ▼  Stapel-Extensions
    - ▼  Parameter-Setter-Extension set file path and file name
      - >  Setze Parameter - ARCHIV\_PFAD
      - >  Setze Parameter - ARCHIV\_DATEINAME
    - ▼  Renderer-Extension render PDF
      - ▼  Metadaten für den Renderer
        - ▼  Metadaten für den Beginn jedes Outputs
          -  Schlüssel-Parameter PDF\_TITLE = PDF\_TITLE
          -  Schlüssel-Parameter PDF\_CREATOR = PDF\_CREATOR
    - ▼  Persistenter-Extension write file to storage
      - ▼  Konfiguration
        - ▼  Dateiname
          - ▼  Dynamische Werte
            -  Dynamischer Wert - 1 - ARCHIV\_DATEINAME
    - ▼  General-File-Extension generate XML for archiving
      - ▼  Dateiname
        - ▬ Parameter - ARCHIV\_DATEINAME

## 7.2.4 EMAIL-Stapel

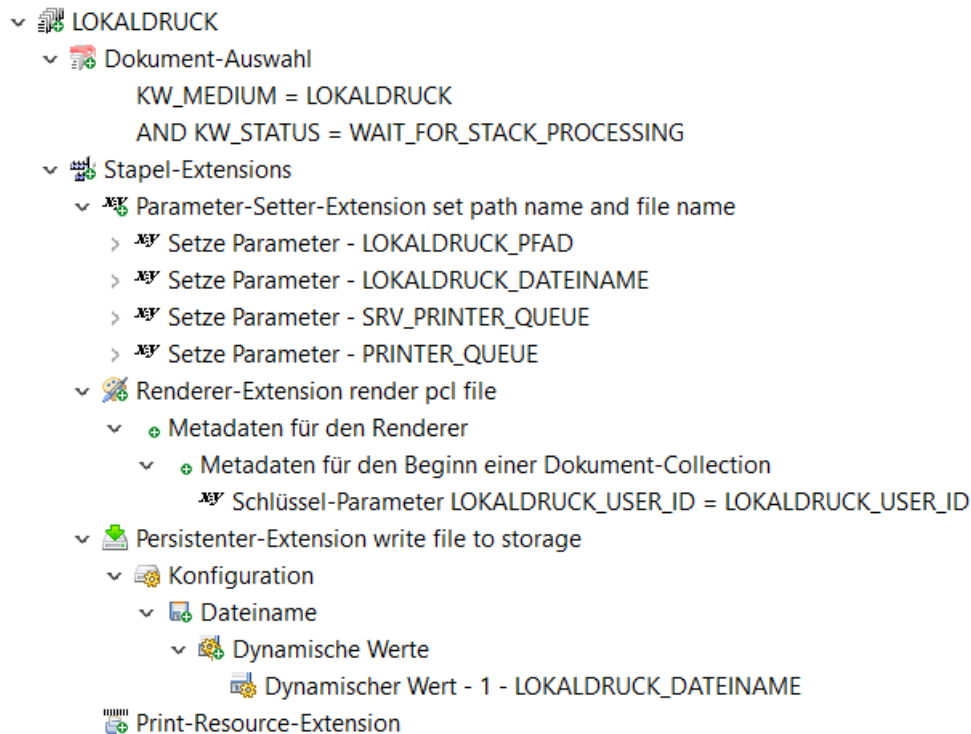
Hier wird je Empfänger eine E-Mail mit dazugehörigen Anhängen generiert.

- ▼  EMAIL
  - ▼  Dokument-Auswahl
    - ▼  KW\_MEDIUM in (EMAIL,EMAILATTACHMENT)
      - Vergleichswert: EMAIL
      - Vergleichswert: EMAILATTACHMENT
      - AND KW\_STATUS = WAIT\_FOR\_STACK\_PROCESSING
  - ▼  Stapel-Extensions
    -  Renderer-Extension Render MAIL Body
    -  Renderer-Extension Render MAIL Attachments
    - ▼  E-Mail-Extension Advanced
      - >  Transport-Konfiguration
      - >  E-Mail-Header
      - >  E-Mail-Text
      - >  Anlagen
      -  Neuer Durchgang
    - ▼  JavaScript-Extension set error status for group
      - ▼  Ereignisse
        - ▼  Ereignis - Bei jedem Dokument
          -  Zu schreibende Parameter
          -  Zu lesende Parameter
          -  Zu schreibende Ressourcen
          -  Zu lesende Ressourcen
  - ▼  Sortierkriterien
    - KW\_INPUT\_ID aufsteigend
    - KW\_MEDIUM aufsteigend
    - KW\_SPLIT\_ID aufsteigend
    - MAIL\_TO aufsteigend
  - ▼  Dokument-Gruppen
    -  Recipient



### 7.2.5 LOKALDRUCK-Stapel

Dieser Stapel erzeugt eine PCL-Druckdatei, versieht diese mit PJP-Kommandos und gibt sie auf einen lokalen Drucker aus. Die Druckerauswahl erfolgt über den Parameter LOKALDRUCK\_DRUCKER, der über den Datenmodellknoten \$Ausgabesteuerung.LOKALDRUCK.Druckername befüllt wird. Der Parameter LOKALDRUCK\_USER\_ID kann z. B. dafür benutzt werden, ein Follow-Me-Printing zu realisieren.



## 8 Adaption des Frameworks an Kundenprojekte

Folgende Änderungen und Erweiterung sind am Framework sind notwendig, um das Starterframework für Kunden anzupassen.

- Anpassen der Datenmodelle Ausgabesteuerung, Absender, Partner, sowie
- Anpassen der Mappings an die Datenquellen des Kunden
- Definition der Teildokumenttypen und Styles gemäß CD und Projektbedarf
- Anpassen der Ausgabesteuerung
- Ausimplementierung der M/OMS-Stapel
- Implementierung der Fachdatenmodelle und -mappings.