Starterframework TONIC

Konzeption und Benutzung

Version: 6.13  
Stand: 01.12.2022

[1 Einführung 4](#_Toc121136788)

[1.1 M/TEXT Version 4](#_Toc121136789)

[1.2 Begriffe 4](#_Toc121136790)

[1.2.1 Auftragsteuerung 4](#_Toc121136791)

[1.2.2 Korrespondenzen 4](#_Toc121136792)

[1.2.3 Exemplare 4](#_Toc121136793)

[1.2.4 Brief 4](#_Toc121136794)

[1.2.5 Anschreiben 5](#_Toc121136795)

[1.2.6 Anlage, Beilage, Beileger 5](#_Toc121136796)

[1.2.7 Sendung 5](#_Toc121136797)

[1.2.8 Kopie, Durchschrift 5](#_Toc121136798)

[1.2.9 Usecases 6](#_Toc121136799)

[1.2.9.1 Eine Korrespondenz mit einem Exemplar 6](#_Toc121136800)

[1.2.9.2 Eine Korrespondenz mit einem Erst-Exemplar und weiteren Exemplaren 6](#_Toc121136801)

[1.2.9.3 Zwei Korrespondenzen mit jeweils einem Exemplar 7](#_Toc121136802)

[1.3 Weitere Begriffe 7](#_Toc121136803)

[1.3.1 Empfänger und alternativer Empfänger 7](#_Toc121136804)

[1.3.2 Absender 7](#_Toc121136805)

[1.3.3 Usecase 8](#_Toc121136806)

[1.3.3.1 Erst-Exemplar mit einem alternativen Empfänger 8](#_Toc121136807)

[1.4 Weitere Begriffe 8](#_Toc121136808)

[1.4.1 Teildokumente – die Bestandteile eines Exemplars 8](#_Toc121136809)

[1.4.2 Teildokumenttypen 8](#_Toc121136810)

[1.4.2.1 Usecase: Unterdrücken von Teildokumenten bei bestimmten Exemplaren 9](#_Toc121136811)

[1.4.3 Ausgabekanäle 9](#_Toc121136812)

[1.4.4 Ausgabesteuerung 9](#_Toc121136813)

[1.4.5 Ausgabeziel 9](#_Toc121136814)

[2 Datenmodelle 10](#_Toc121136815)

[2.1 Auftragssteuerung.datamodel 10](#_Toc121136816)

[2.2 Korrespondenz.datamodel 11](#_Toc121136817)

[2.3 Exemplar.datamodel 12](#_Toc121136818)

[2.4 Datenmodelle Absender und Empfaenger 13](#_Toc121136819)

[2.5 Ausgabesteuerung.datamodel 13](#_Toc121136820)

[2.6 Datenmapping: Anforderungen an die Datenquelle 13](#_Toc121136821)

[3 Aufgaben des Frameworks 14](#_Toc121136822)

[3.1 Exkurs Inhaltserweiterungen 14](#_Toc121136823)

[3.2 Inhaltserweiterungen im Framework 16](#_Toc121136824)

[3.3 Auftragssteuerung: Korrespondenzen und Exemplare 18](#_Toc121136825)

[3.3.1 Korrespondenz-Baustein 18](#_Toc121136826)

[3.3.2 Alle Exemplare in der M/Workbench anzeigen 18](#_Toc121136827)

[4 Beispiel-Teildokumenttypen 19](#_Toc121136828)

[4.1 Allgemeine Layoutelemente 20](#_Toc121136829)

[4.2 Brief-Teildokument 20](#_Toc121136830)

[4.2.1 Frameworkbausteine im Brief-Teildokument 22](#_Toc121136831)

[4.3 AGB-Teildokument 23](#_Toc121136832)

[4.4 Anlage-Teildokument 25](#_Toc121136833)

[4.5 Infoblatt-Teildokument 27](#_Toc121136834)

[4.6 Rückantwort-Teildokument 29](#_Toc121136835)

[5 Styles 31](#_Toc121136836)

[6 Beispielvorlagen 32](#_Toc121136837)

[7 M/OMS Ansteuerung 33](#_Toc121136838)

[7.1 Baustein Ausgabesteuerung 33](#_Toc121136839)

[7.1.1 ZENTRALDRUCK 33](#_Toc121136840)

[7.1.2 ARCHIV 33](#_Toc121136841)

[7.1.3 LOKALDRUCK 34](#_Toc121136842)

[7.1.4 FAX 34](#_Toc121136843)

[7.1.5 EMAIL oder EMAILATTACHMENT 35](#_Toc121136844)

[7.2 M/OMS Konfiguration 36](#_Toc121136845)

[7.2.1 ZENTRALDRUCK-Stapel 37](#_Toc121136846)

[7.2.2 ZENTRALDRUCK\_PDFVT-Stapel 38](#_Toc121136847)

[7.2.3 ARCHIV-Stapel 39](#_Toc121136848)

[7.2.4 EMAIL-Stapel 40](#_Toc121136849)

[7.2.5 LOKALDRUCK-Stapel 41](#_Toc121136850)

[8 Adaption des Frameworks an Kundenprojekte 41](#_Toc121136851)

# 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 bespielhafte Umsetzung einer M/OMS Ansteuerung zu finden.

Das Projekt **Framework\_Demo\_DE** zeigt Beispielvorlagen, welche die Grundfunktionen der Dokumentensteuerung beinhalten. Auf einige dieser Beispielvorlagen 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.

## 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.

## Begriffe

### 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.

### 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.

### 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.

### Brief

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

### Anschreiben

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

### Anlage, Beilage, Beileger

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

### Sendung

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

### 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.

### Usecases

#### Eine Korrespondenz mit einem Exemplar

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

Korrespondenz

Exemplar

-----------------

-------------

---------

Sehr geehrter Herr Müller,

wir übernehmen die Kosten der Behandlung.

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.

#### 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 Kenntnisnahme in Kopie. Der Arzt ist Empfänger des Erstexemplars. Der Patient ist Empfänger des weiteren Exemplars.

Korrespondenz

Erst-Exemplar

-----------------

-------------

---------

Sehr geehrter Herr Dr. Friedrich,

für Ihren Patienten Hr. Müller übernehmen wir die Kosten der Behandlung.

Weiteres Exemplar

-----------------

-------------

---------

Sehr geehrter Herr Müller,

Sie erhalten eine Kopie des Schreibens an Hr. Dr. Friedrich …



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.

#### 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.

Korrespondenz

Erst-Exemplar

-----------------

-------------

---------

Sehr geehrter Herr Dr. Friedrich,

für Ihren Patienten Hr. Müller übernehmen wir die Kosten der Behandlung.

Korrespondenz

Erst-Exemplar

-----------------

-------------

---------

Sehr geehrter Herr Müller,

wir übernehmen die Kosten der Behandlung.

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

## Weitere Begriffe

### 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.

### 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.

### Usecase

#### 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.

Korrespondenz

Erst-Exemplar

-----------------

-------------

---------

Sehr geehrter Frau Schmidt,

Sie erhalten dieses Schreiben für Herrn Müller.

-----------------

-------------

---------

Sehr geehrter Herr Müller,

wir übernehmen die Kosten der Behandlung.

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

## Weitere Begriffe

### 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.

### Teildokumenttypen

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 Teildokumenttypen und deren Nutzung in Vorlagen.

#### 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.

INFOBLATT

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna

------------------

-----------

------------

Sehr geehrter Herr Dr. Friedrich,

für Ihre Patientin Frau Schmidt übernehmen wir die Kosten der …

------------------

-----------

------------

Sehr geehrte Frau Schmidt,

im Anhang erhalten Sie eine Kopie des Schreibens, welches wir an Hr. Dr. Friedr

Weiteres Exemplar

Erst-Exemplar

------------------

-----------

------------

Sehr geehrter Herr Dr. Friedrich,

für Ihre Patientin Frau Schmidt übernehmen wir die Kosten der …

Korrespondenz

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.

### 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.

### 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.

### 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.

# Datenmodelle

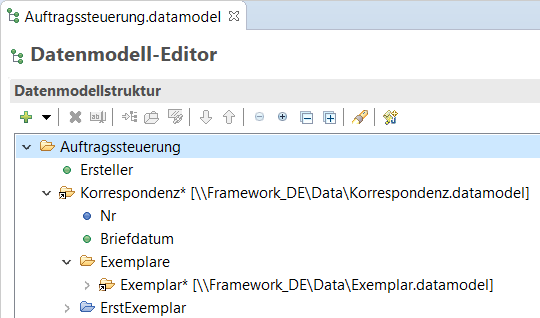
Nachfolgend werden die wichtigsten Datenmodelle beschrieben und deren Aufgaben erläutert. Details zum Aufbau der einzelnen Datenmodelle kann man jederzeit im Framework nachsehen. Alle Framework-Datenmodelle befinden sich im Projekt Framework im Verzeichnis Data.

## 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.



## Korrespondenz.datamodel

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

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## Exemplar.datamodel

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

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

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.

## Datenmodelle Absender und Empfaenger

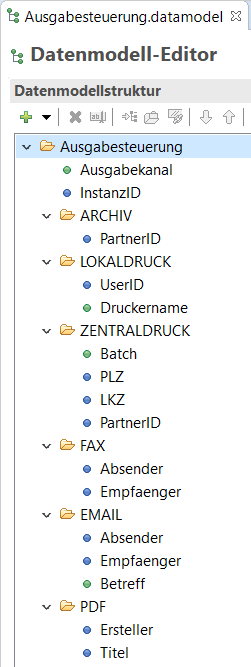
Diese Datenmodelle werden hier nicht explizit beschrieben da ihre Struktur den kundenindividuellen Gegebenheiten angepasst werden muss.

## 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.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## 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.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

# Aufgaben des Frameworks

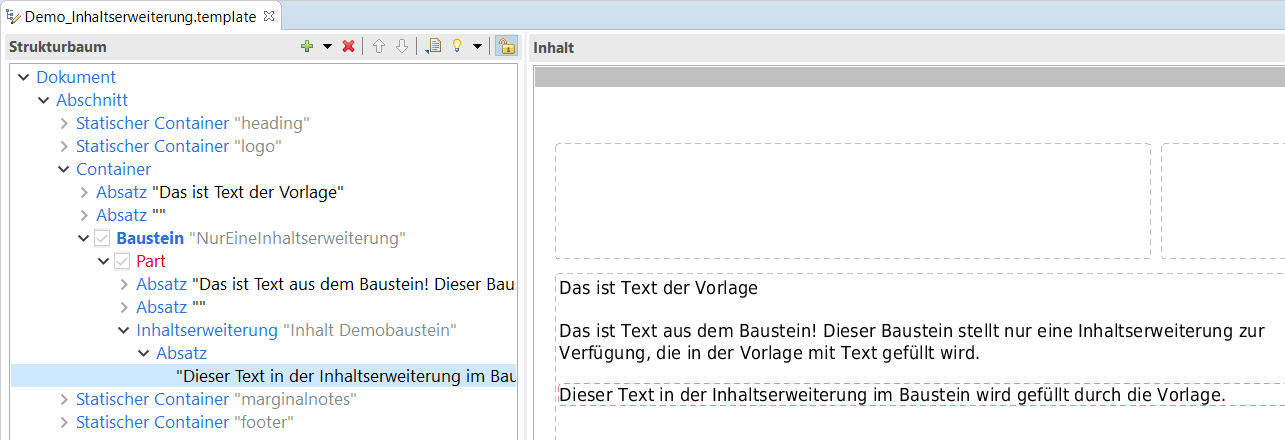
Das Framework kapselt die gesamte Logik der Auftragssteuerung, die zur Erzeugung der Korrespondenzen und Exemplare notwendig ist. Ein Vorlagenentwickler kann sich so auf die Erstellung seiner Vorlage konzentrieren und benötigt kein Wissen über diese Logik. Diese Auftragssteuerung ist weiter unten im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

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

## 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>

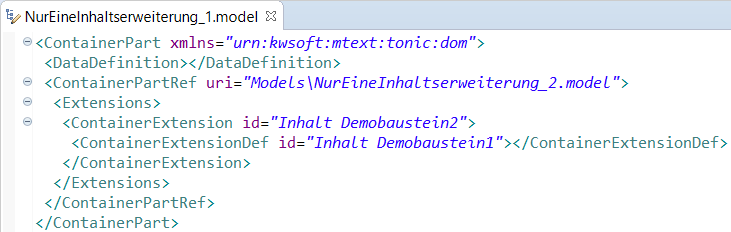
</Container>

</Section>

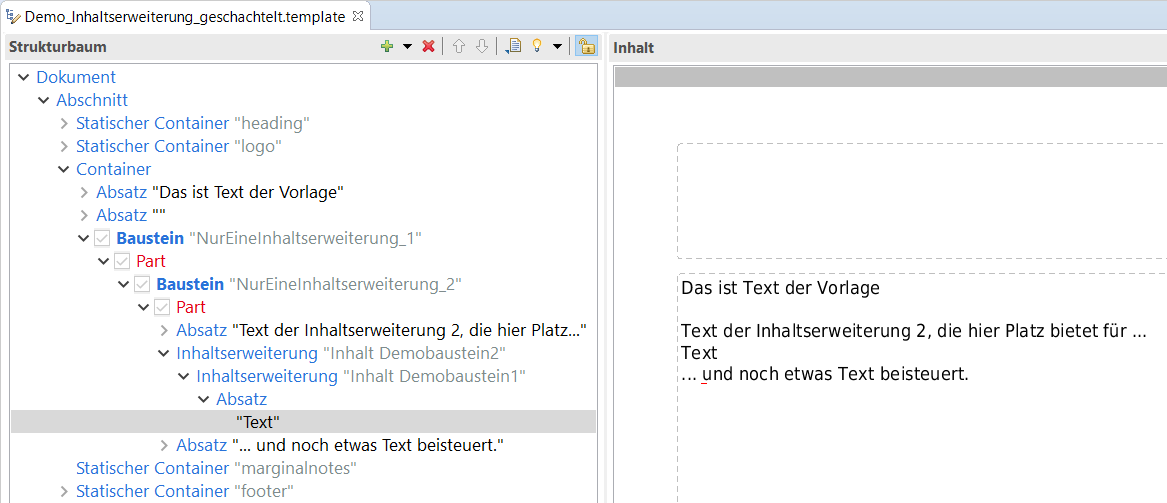
</Document>

</RootPart>

Inhaltserweiterungen können geschachtelt werden. Dazu wurde die Beispielvorlage 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.



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



## 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.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

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.

Vorlage

Korrespondenz.model

Korrespondenz Inhalt

Brief.model

Brief Inhalt

Hier steht der Inhalt des Anschreibens

AGB.model

AGB Inhalt

Hier steht der Inhalt der ABG

## Auftragssteuerung: Korrespondenzen und Exemplare

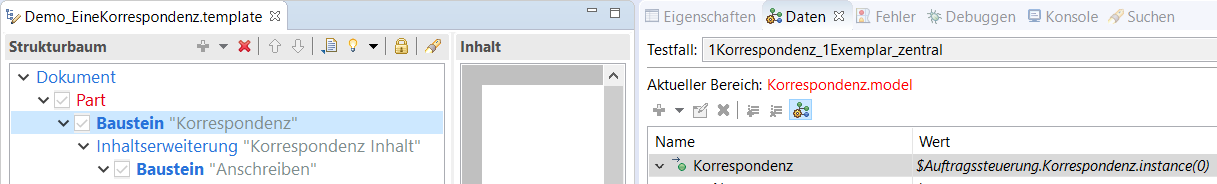
### Korrespondenz-Baustein

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.



### 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.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

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.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

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

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Darüber hinaus werden Platzhalter für Steuerzeichen am Blattrand ausgegeben.

# Beispiel-Teildokumenttypen

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

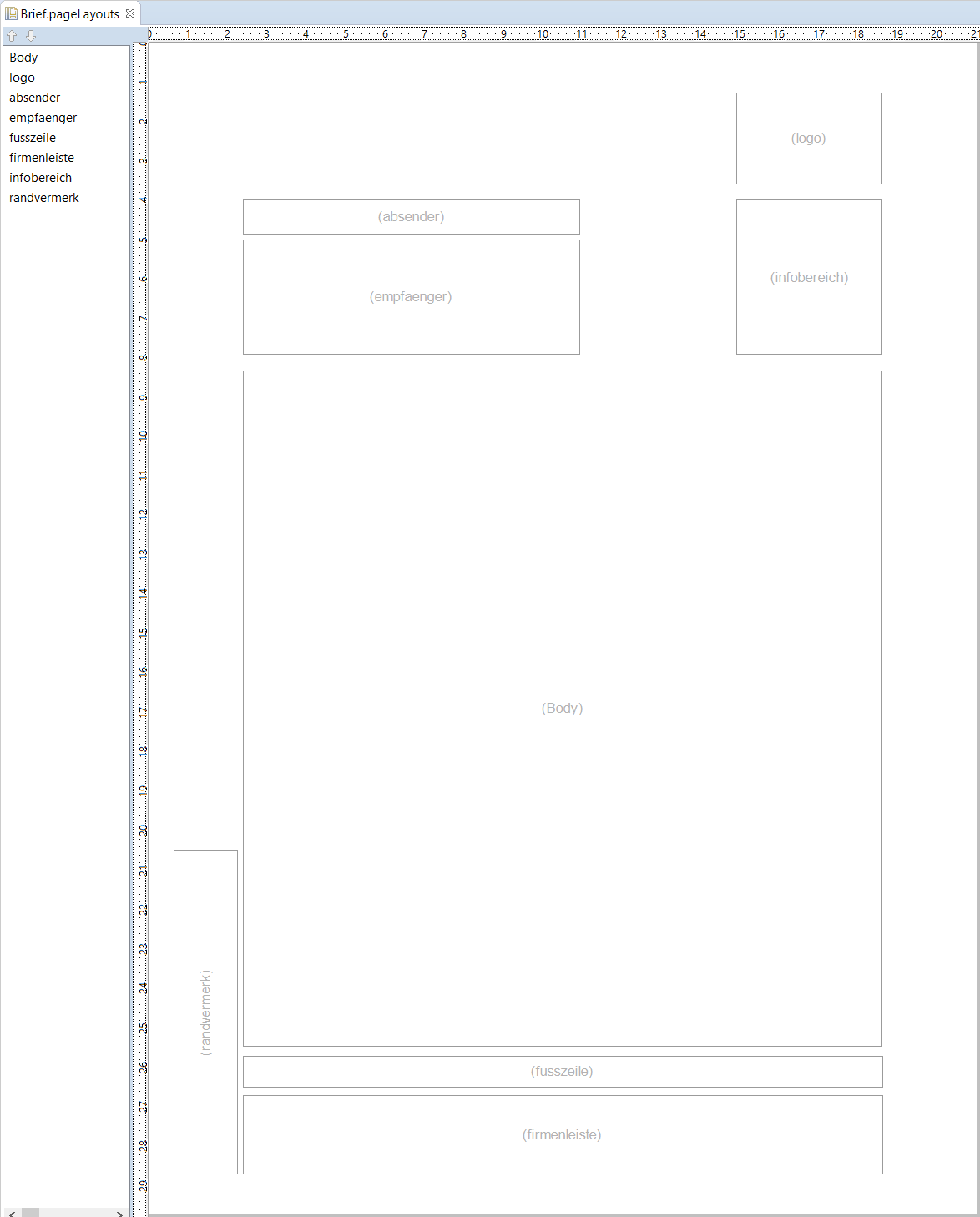
* Brief
* AGB
* Anlage
* Infoblatt
* Rückantwort

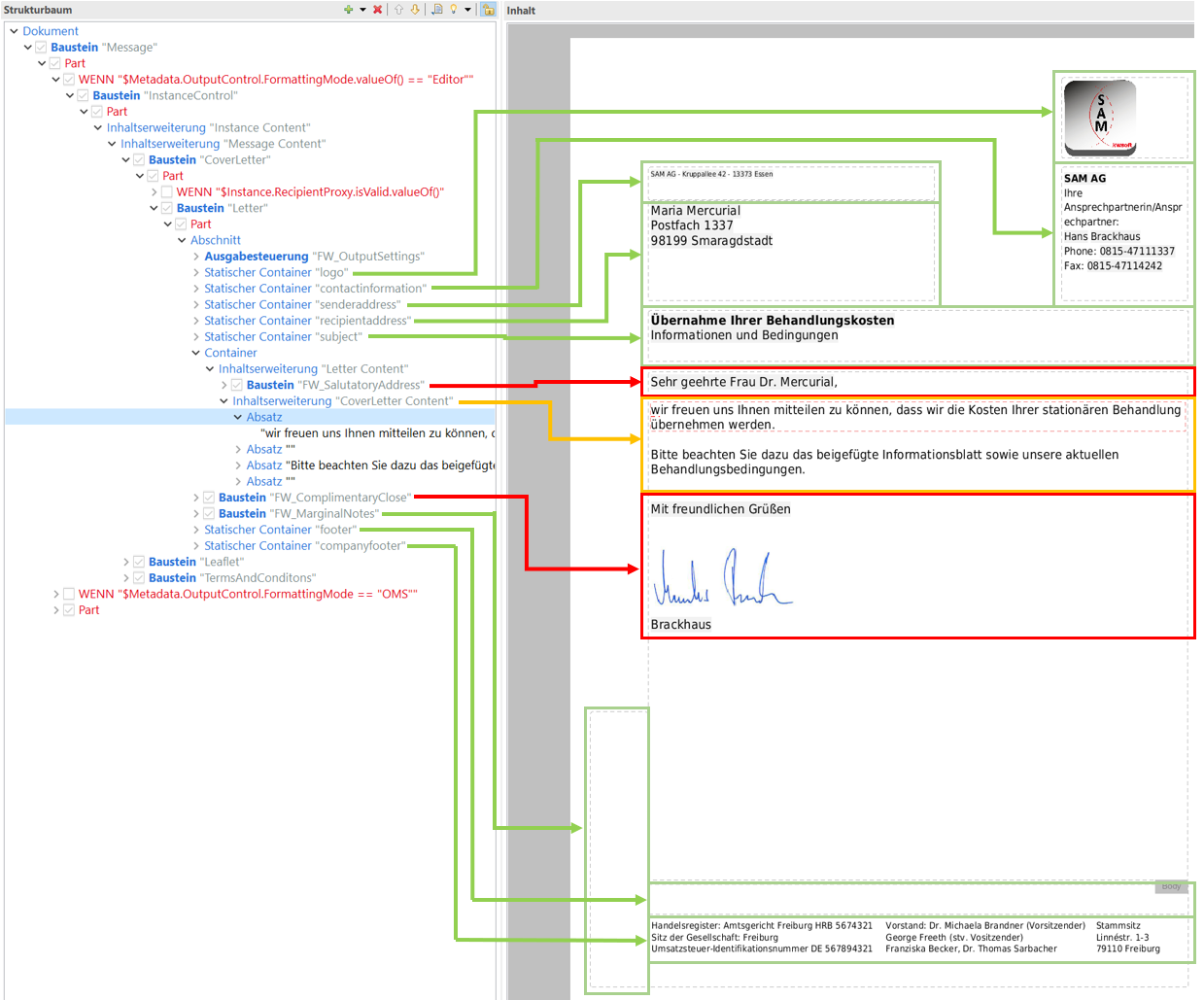
## Allgemeine Layoutelemente

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.

## 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.

### Frameworkbausteine im Brief-Teildokument

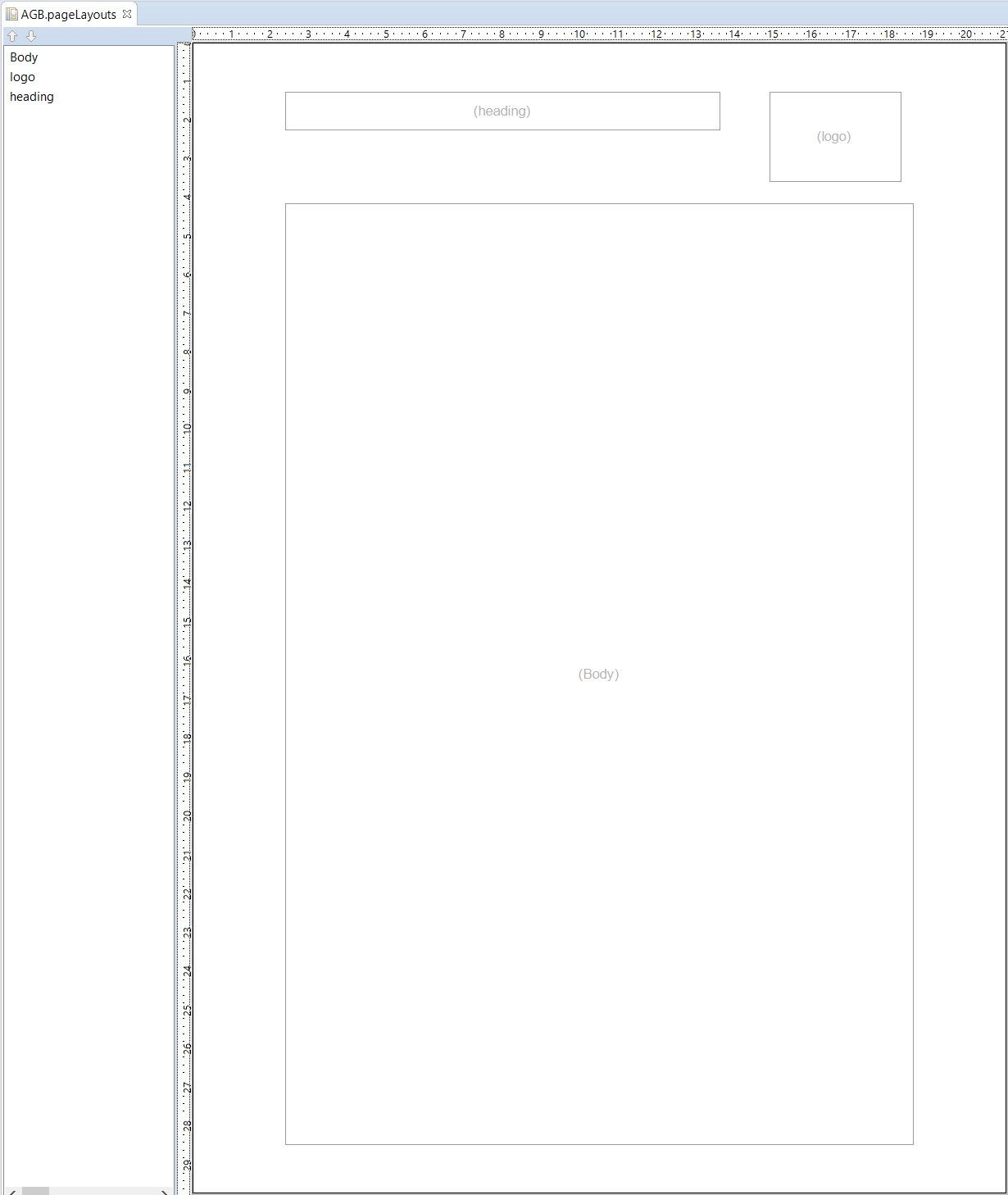
|  |  |  |
| --- | --- | --- |
| Baustein | Beschreibung | Parameter |
| FW\_Logo.model | Der Bereich rechts oben stellt ein Logo bereit | keine |
| FW\_Absender.model | Brief-Absender oberhalb der Anschrift | Absender.datamodel |
| FW\_Empfaenger.model | Brief-Anschrift | Empfaenger.datamodel |
| FW\_Betreff.model | 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 |

## AGB-Teildokument

Das AGB-Teildokument stellt ein einfaches Layout zur Verfügung. Als einziger Parameter wird der Titel übergeben. Der Inhalt wird in der Vorlage über die Inhaltserweiterung „AGB Inhalt“ gefüllt. In der Beispielvorlage ist dies zweispaltig realisiert.

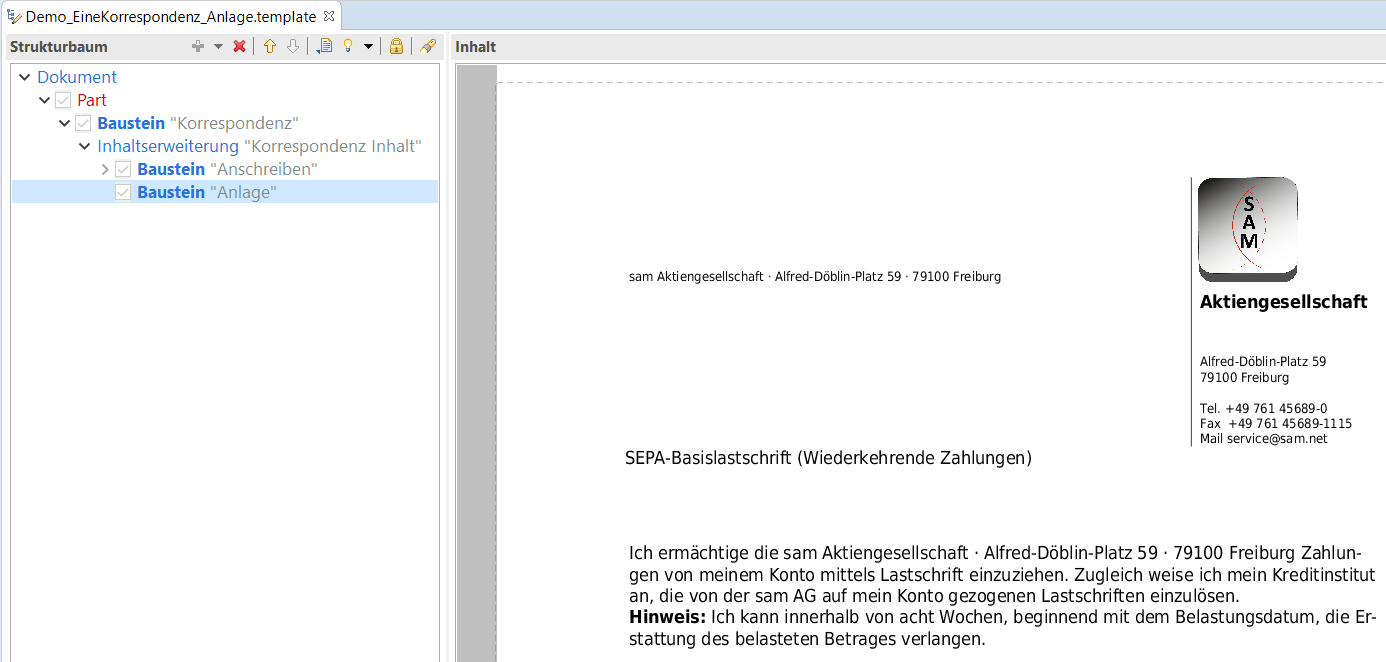
Ein Bild, das Text enthält.

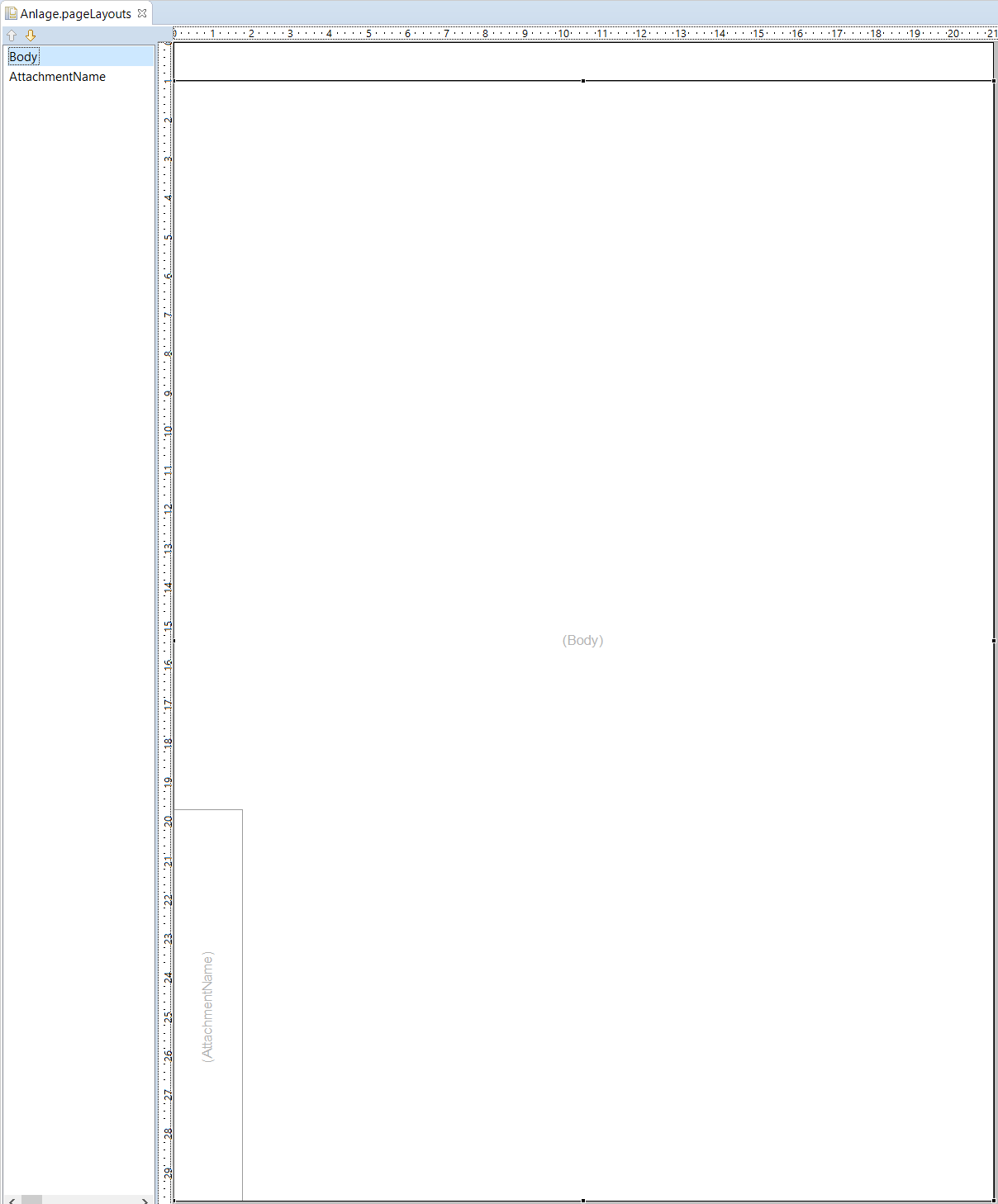
Automatisch generierte Beschreibung



## 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.



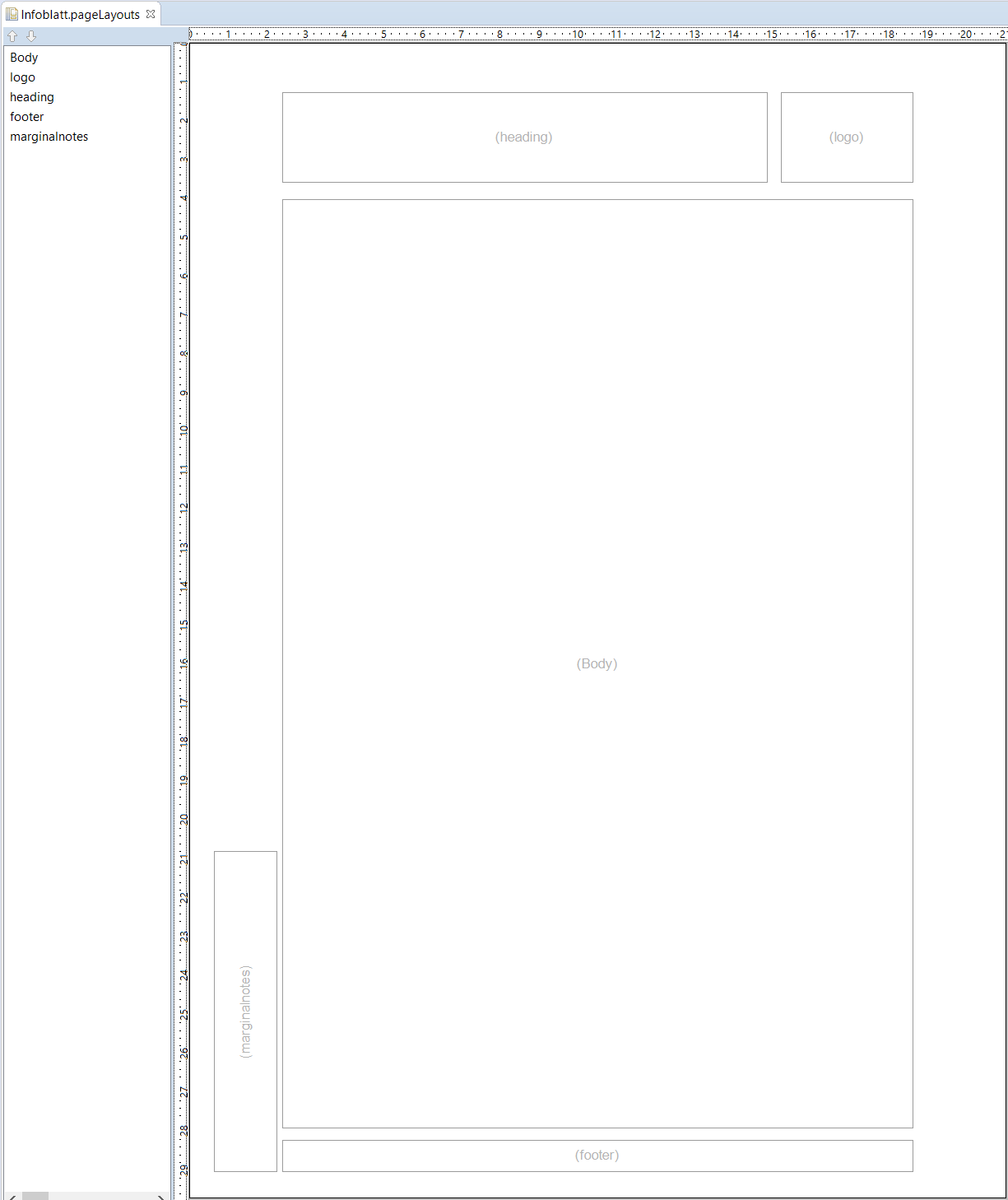


## Infoblatt-Teildokument

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.

Ein Bild, das Text enthält.

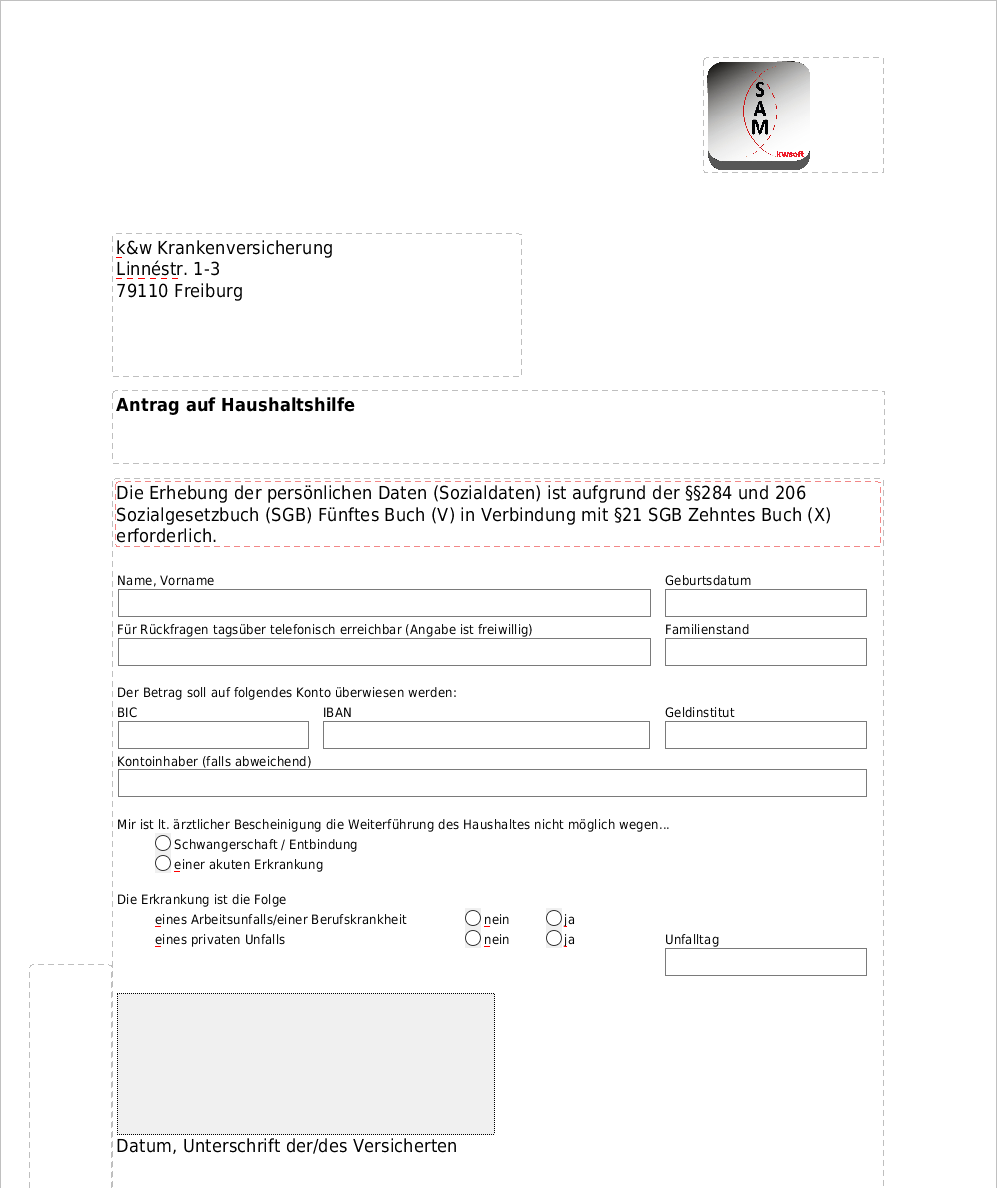
Automatisch generierte Beschreibung

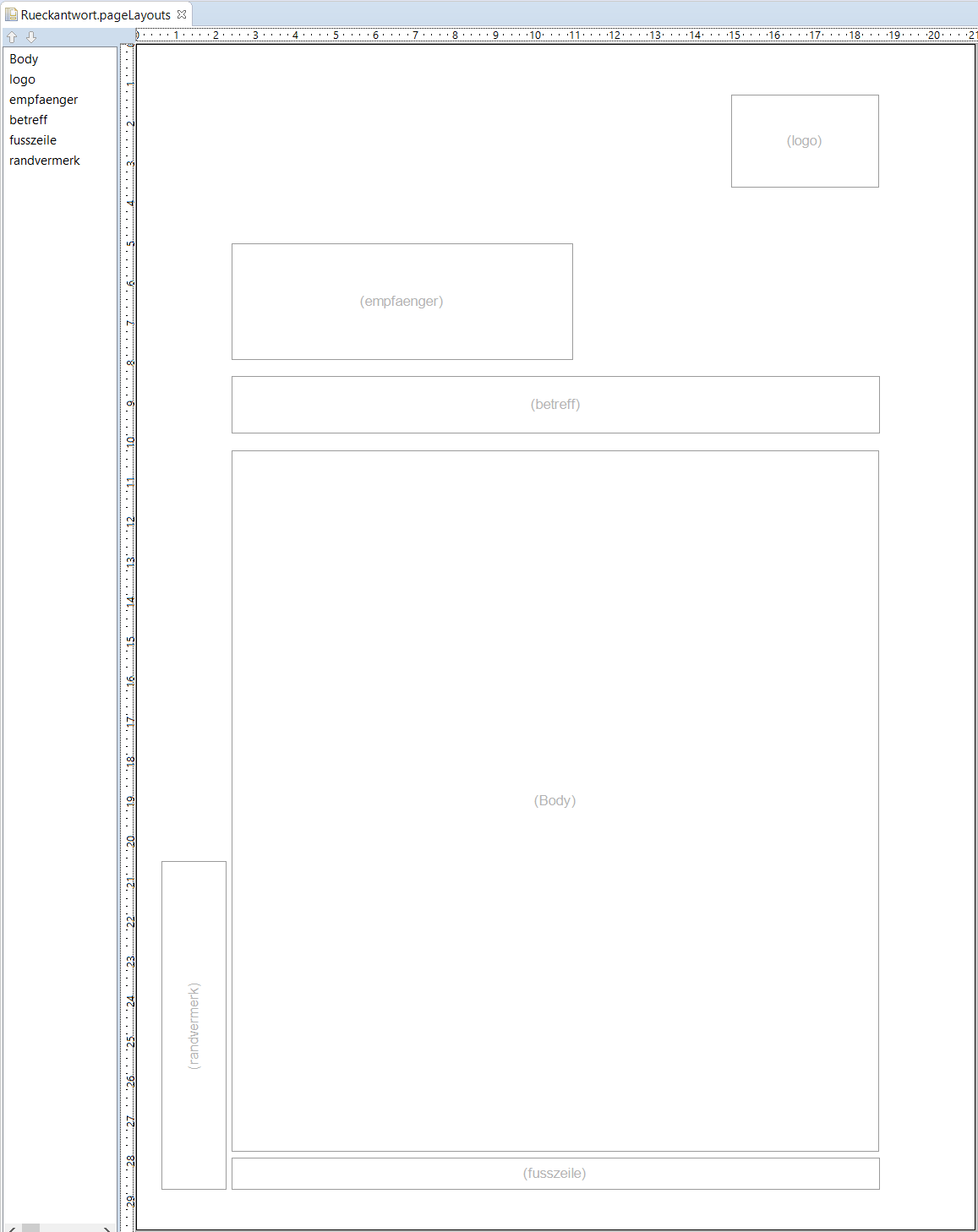


## Rückantwort-Teildokument

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.





# 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.

# 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. |

# M/OMS Ansteuerung

## 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.

### ZENTRALDRUCK

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

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ist dieser gefunden, werden die OMS-Parameter gesetzt:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### ARCHIV

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

Ein Bild, das Text enthält.

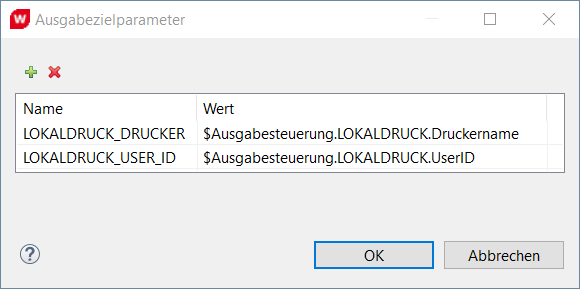
Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

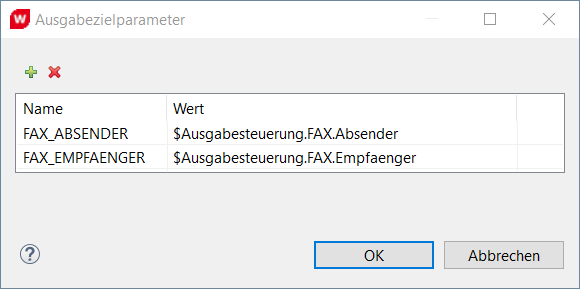
### LOKALDRUCK

Siehe ZENTRALDRUCK, nur mit folgenden OMS-Parametern.



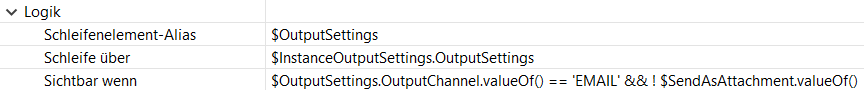
### FAX

Siehe ZENTRALDRUCK, nur mit folgenden OMS-Parametern.



### 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.



Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## 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.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### 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).

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### ZENTRALDRUCK\_PDFVT-Stapel

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

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### ARCHIV-Stapel

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

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### EMAIL-Stapel

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

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### LOKALDRUCK-Stapel

Dieser Stapel erzeugt eine PCL-Druckdatei, versieht diese mit PJL-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.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

# 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.