

## Word und Excel-Dokumenterzeugung mit Posity

### Inhalt

1. Dokumentengenerierung mit Posity.....	2
2. Syntaxnotation .....	3
3. Syntax und Semantik der Platzhalter .....	4
4. Beispiel Word-Vorlage und erzeugtes Dokument .....	13
5. Beispiel Excel-Vorlage und erzeugtes Dokument .....	21

# 1. Dokumentengenerierung mit Posity

Dieses Dokument beschreibt, wie mittels Posity individuelle und anspruchsvolle Word- und Excel-Dokumente erzeugt werden können. Zum Erzeugen eines Dokumentes muss zunächst eine Vorlage in Word (.docx), respektive Excel (.xlsx) erstellt werden. Dabei kann das Layout mit den Mitteln von Word, resp. Excel frei gestaltet werden. In diese Vorlage können dann Platzhalter (z.B. für den Namen des Kunden) eingefügt werden, welche durch Posity anschliessend ersetzt werden. Die **Platzhalter** sind dabei immer in geschweiften Klammern anzugeben (z.B. '{Partner.CustomerName}'). Posity ersetzt in der Verarbeitung dann diese Platzhalter mit den konkreten Werten. In Excel können in der Vorlage auch mehrere Sheets der Arbeitsmappe verwendet werden.

**Definition Begriffe:** Die Begriffe *Liste*, *Sheet*, *Tabelle* werden in den folgenden Ausführungen wie folgt verstanden:

1. **Tabelle:**  
Mit dem Begriff Tabelle sprechen wir die Input-Daten an, welche durch Posity verarbeitet werden – nicht eine Word- oder eine Excel-Tabelle.
2. **Liste und Sheet in Word:**  
Wird der Begriff Liste in Zusammenhang mit Word verwendet, ist damit eine 'normale' Word-Tabelle angesprochen (gefüllt mit Posity-Platzhaltern). Es handelt sich dabei nicht um eine in Word integrierte Excel-Tabelle. Der Begriff Sheet hat hier keine Bedeutung.
3. **Liste und Sheet in Excel:**  
Das Sheet bezeichnet ein Blatt innerhalb einer Excel-Arbeitsmappe. Dieses Sheet wird häufig auch als Tabelle bezeichnet – was hier vermieden wird (die Tabelle enthält die Input-Daten, siehe 1. Punkt). Unter Liste verstehen wir einen zusammenhängenden Teil von Zeilen innerhalb eines Sheets (z.B. Zeilen 5 bis 12), welche mit Daten befüllt werden soll. Jede Zeile einer solchen Liste ist durch die Platzhalter {Row}, {FooterBreak}, {HeaderBreak} oder {Spacer} (deren Wirkung wird später erklärt) in der ersten Spalte gekennzeichnet. Ein einzelnes Excel-Sheet kann daher mehrere Listen beinhalten.

**Platzhalter in Listen:** Mittels weiteren Platzhaltern (z.B. {DecimalSeparator=,} zum Festlegen des Dezimaltrennzeichens) kann das Layout des Dokuments gesteuert werden. Gewisse Platzhalter nehmen auch Einfluss auf das Layout von Listen. Diese Platzhalter können in Excel-Vorlagen direkt eingesetzt werden. In Word-Dokumenten müssen diese Platzhalter zwingend innerhalb von Listen verwendet werden.

**Platzhalter in Arbeitsblattname:** (nur Excel) Können mit folgenden Einschränkungen verwendet werden:

1. Der Platzhalter darf nicht länger als 31 Zeichen sein (Beschränkung von Excel)
2. Der für den Platzhalter eingesetzte Wert wird bei 28 Zeichen abgeschnitten

**Tags von Platzhaltern:** Bestimmte Platzhalter können durch ein oder mehrere Tags (Angaben) ergänzt werden. Diese Tags müssen immer mittels | (Pipe) voneinander getrennt werden, z.B.

{CustomerOffer.CashDiscount|Format:"Skonto: "##,##0.00"% " |Zero:}.

**Serienbrieffunktion:** Es können auch Dokumente erzeugt werden, welche mehrere Input-Tabellen benötigen. So können in einem Dokument z.B. die Input-Tabellen Kunde, Offerte und Offertpositionen verarbeitet werden. Dabei ist es auch möglich, für mehrere Kunden gleichzeitig deren Offerten und Offertpositionen in 'einzelnen' Dokumenten (z.B. ein Dokument je Offerte) zu erzeugen (ähnlich der Serienbrieffunktion von Word).

**Feldfunktionen:** In den erzeugten Word-Dokumenten können auch Word-Feldfunktionen verwendet werden (siehe auch Platzhalter {UpdateFields}). Innerhalb der Feldfunktionen können Platzhalter eingesetzt werden, die dann nach Erzeugung des Dokumentes angewendet werden sollen, z.B. {IF "{Question.Selected}"="True" "☒" "☐"} (fette geschweifte Klammern sind Teil der Word-Feldfunktion).

**Schutz vor Änderungen:** Möchten Sie das erzeugte Word-Dokument vor Änderungen schützen, können Sie die Vorlage mit einem Passwort vor Änderungen schützen. Das erzeugte Dokument übernimmt den Schreibschutz, sowie das zugehörige Passwort in das erzeugte Dokument.

## 2. Generelle Regeln der Syntaxnotation

Folgende Syntax-Regeln müssen immer eingehalten werden:

- Leerschläge (Spaces) sind NICHT erlaubt, ausser diese sind in Formatdefinitionen explizit erwünscht.
- Gross/Kleinschreibung muss beachtet werden
- Innerhalb von Platzhaltern können keine Steuerzeichen (Paragraph ¶, NewLine ↵, TabChar →) verwendet werden. Hierzu müssen die Platzhalter {NewParagraph}, {NewLine} und {TabChar} verwendet werden (siehe Nicht-Terminal-Symbol stringWithBreak und «3. Syntax und Semantik der Platzhalter»).

Zeichenerklärung zur Syntaxnotation:

- ::= definiert einen Platzhalter
- [ ] der innere Teil ist optional
- { } der innere Teil kann beliebig oft verwendet werden (auch kein Mal)
- ( ) Gliederung zur Abgrenzung oder zur besseren Verständlichkeit der Definition – dürfen aber nicht in die Vorlage übernommen werden (jeweils in oranger Farbe)
- | Auswahl (oder); Text vor oder hinter dem Pipe muss verwendet werden; werden mehrere Optionen kombiniert (z.B. {CO.CashDiscount|Format:"Skonto:###,##0.00"% "|Zero:}, dann werden die Optionen in der Reihenfolge der Erwähnung angewendet.
- ";" der innere Teil (hier der Strichpunkt) muss zwingend als Textkonstante in die Vorlage übernommen werden (jeweils in blauer Farbe)
- <> der Inhalt dieses Symbols (Nicht-Terminal-Symbol) wird für die Syntax (nächstes Kapitel) an anderer Stelle definiert, in der unmittelbar folgende Liste unten:

Folgende Nicht-Terminal-Symbole (diese Symbole kommen nicht mehr als «Worte» in den erzeugten, endgültigen Platzhaltern vor) werden in der nachfolgenden Syntaxnotation für die Definition der Platzhalter verwendet:

- AbsolutPathFileAndExtension ::= [][{"< Attribute >"}][] Pfadangabe gemäss Windows-Standard
- AnyString ::= Beliebige Zeichenkette inklusive Sonderzeichen aber ohne { und }.
- Attribute ::= (<TableName>|<TableAliasName>)."<AttributeName>
- AttributeName ::= <String>
- AttributeWithNr ::= ("{"< Attribute >"}")|<Number> Attribute value must contain a number
- Bold ::= <Boolean>
- Boolean ::= "true" | "false"
- Color ::= "0".."255" Ganzzahlige Werte zwischen 0 und 255
- ColorB ::= Color
- ColorG ::= Color
- ColorR ::= Color
- ComparisonOperator ::= "<=" | ">=" | "<" | ">" | "<>" | "="
- Condition ::= ("{"<Attribute>"}"|"{"<String>"|"<Number>"}<ComparisonOperator>("{"<Attribute>"}"|"{"<String>"|"<Number>")
  - Die Datentypen des Vergleichs müssen übereinstimmen.
- FontName ::= <String> Gültige Bezeichnung einer Schriftart.
- FontSize ::= <Number>
- FormatPattern ::= <AnyString>
  - Hier sind die Formatierungsregeln gemäss Spezifikation des Ausgabeformat-Format-Befehls von C# erlaubt.
  - In der Formatangabe können beliebige Zeichen verwendet werden, also auch Tabulatoren, Linien- und Seitenumbrüche (die Wirkung ist abhängig davon, ob es eine Word- oder Excel-Vorlage ist). Achtung: Falls verwendete Zeichen fälschlicher Weise als Teil der Formatierungsangabe interpretiert werden können, muss ein Escape-Zeichen ("\") vorangestellt werden.

- Dezimaltrennzeichen müssen immer als Punkte, Tausendertrennzeichen als Komma geschrieben werden (siehe auch Platzhalter `DecimalSeparator` und `ThousandSeparator`).
- Bei Zahlen kann entweder ein einzelnes Format angegeben werden, oder es können zwei Formate angegeben werden (für die positive Zahl und die negative Zahl), oder es können drei Formate angegeben werden (für die positive Zahl, für die negative Zahl und für den 0-Fall). Die verschiedenen Fälle werden mittels Strichpunkt (;) getrennt.

Enthält die Formatierung Textkonstanten vor oder hinter der Zahl (wie im Beispiel unten) und die Zahl kann negative Werte annehmen, dann muss die Formatierung immer auch für den negativen Wert angegeben werden, da ansonsten das Minus-Zeichen vor die erste Textkonstante platziert wird. Beispiele:

- Eine Formatangabe: `"Temperatur: "#,##0.00" C°"`
- Zwei Formatangaben: `"Temperatur: "#,##0.00" C°";"Temperatur: "-#,##0.00" C°"`. Beachten Sie, dass die GESAMTE Formatangabe vor und nach dem Strichpunkt wiederholt werden muss.
- Drei Formatangaben: `"Temperatur: "#,##0.00" C°";"Temperatur: "-#,##0.00" C°";"*keine*"`. Beachten Sie, dass die GESAMTE Formatangabe vor und nach den Strichpunkten wiederholt werden muss.
- Bei einer Dauer (Duration) können die Zeiten auch im Industrieformat ausgegeben werden. Hierzu muss vor das eigentliche Format ein "[D]" (Tage), "[H]" (Stunden), "[m]" (Minuten) oder "[s]" (Sekunden) gestellt werden. Ein gültiges Format ist z.B. `"[H]##00.00"` (ergibt z.B. 08.75 Stunden). Kann nicht in Excel-Vorlagen verwendet werden.
- `Italic` ::= `<Boolean>`
- `Level` ::= `<Digit>` Werte zwischen 0 und 8 sind erlaubt.
- `Number` ::= `<Digit>["."<Digit>]`
- `Digit` ::= `("0" | .. | "9") {"0" | .. | "9"}`
- `Pixels` ::= `<Digit>`
- `String` ::= `("a" | .. | "9") {"a" | .. | "9"}`
- `StringWithBreak` ::= `{<AnyString> | <NewParagraph> | <NewLine> | <TabChar> | <FormatParagraph>}`
- `SymbolShortcut` ::= `"o" | "O" | "oo" | "OO" | ">" | "<" | "[]"`
- `SymbolUnicode` ::= `<Digit>[<Digit>][<Digit>][<Digit>][<Digit>]` Erlaubt sind Werte für die Unicode-Zeichen gemäss Symbol-Tabelle von Word.
- `TableAliasName` ::= `<TableName>` `TableAliasName` muss mithilfe `TableAlias`-Zuweisung definiert werden.
- `TableName` ::= `<String>`
- `Underscore` ::= `<Boolean>`

### 3. Syntax und Semantik der Platzhalter

```
AttributeValue ::=
  "{ "<Attribute>
  ["|ColorCondition:"<Condition>";"<ColorR>","<ColorG>","<ColorB>]
  ["|FontCondition:"<Condition>";"<FontSize>","<Bold>","<Italic>","<Underscore>]
  ["|DependOn:<Attribute>]
  ["|Format:"[""<StringWithBreak>""]<FormatPattern>[""<StringWithBreak>""]]
  ["|MaxWidth:"<Pixels>]
  ["|MaxHeight:"<Pixels>]
  ["|FitRow:"("Yes"|"No")]
  ["|OldValue"]
  ["|NonZero:"<StringWithBreak>]
  ["|Zero:"<StringWithBreak>]
  ["|True:"<StringWithBreak>]
  ["|False:"<StringWithBreak>]
  "}"
```

- Der Platzhalter wird durch den Wert in den zu verarbeitenden Daten ersetzt, z.B. wird `{CustomerInvoice.InvoiceDate}` durch das Datum der Rechnung (z.B. 10.10.2016) ersetzt.
- Position des Platzhalters: An der Stelle in der Vorlage, an der der Wert stehen soll.

- Der eingefügte Wert des Attributes muss der formalen Definition des nicht Nicht-Terminal-Symbols `<StringWithBreak>` entsprechen.
- **ColorCondition:** Ist die Bedingung erfüllt, wird das Attribut in der entsprechenden Farbe (RGB) angezeigt, z.B. `{CustomerInvoice.InvoiceDate|ColorCondition:{P.NamePartner}="Posity AG";255,50,0}`. Im Vergleich können nur Attribute des aktuellen Datensatzes verwendet werden. Beide Attribute müssen vom Typ String (alle Typen ausser Numeric werden als String eingesetzt) oder vom Typ Numeric sein. Ist der Vergleich nicht möglich (falsche Typen, falsches Attribut) wird die Bedingung ignoriert. Gross/Kleinschreibung wird bei der Auswertung der Vergleichsoperation ignoriert! Attribute müssen in geschweifte Klammern gesetzt werden.
- **FontCondition:** Ist die Bedingung erfüllt, wird das Attribut mit den entsprechenden Angaben formatiert, z.B. `{CustomerInvoice.InvoiceDate|ColorCondition:{P.NamePartner}="Posity AG";10,true,false,false}`. Im Vergleich können nur Attribute des aktuellen Datensatzes verwendet werden. Beide Attribute müssen vom Typ String (alle Typen ausser Numeric werden als String eingesetzt) oder vom Typ Numeric sein. Ist der Vergleich nicht möglich (falsche Typen, falsches Attribut) wird die Bedingung ignoriert. Gross/Kleinschreibung wird bei der Auswertung der Vergleichsoperation ignoriert! Attribute müssen in geschweifte Klammern gesetzt werden.
- **DependOn:** Ist der Wert des hier referenzierte Attributes 0 oder leer, wird das Attribut unterdrückt – ein leerer String ausgegeben, z.B. `DependOn:CustomerInvoice.CashDiscount`.
- **Format:** Im `<FormatPattern>` kann das Ausgabeformat gemäss Spezifikation des Format-Befehls von C# gesetzt werden, ergänzt um einen vorhergehenden und nachfolgenden String (z.B. `"Datum: "dd.MM.yyyy'`, oder `'#,##0.00'` oder `'#,##0' \%`). Siehe auch Beschreibung zu `<FormatPattern>`.
- **MaxWidth** und **MaxHeight** sind nur für Bilder möglich! Überschreitet das Bild eine der angegebenen Grössen wird es proportional verkleinert, bis es die angegebene Maximalgrösse (Seitenverhältnis wird beibehalten) hat. Es kann auch nur eine Grössenbeschränkung verwendet werden. Die Einheit ist in DPI (100 entspricht daher 2.54 cm). Z.B. stellt `MaxWidth:400` sicher, dass das Bild maximal 400 DPI, resp. 5.16cm breit ist, `MaxWidth:400|MaxHeight:200` stellt sicher, dass das Bild maximal 400 DPI breit und maximal 200 DPI hoch ist. Im Word wird der Mechanismus zur Positionierung des Bildes durch die Vorlage gesteuert, im Excel wird das Bild so platziert, dass die Ecken oben links des Bildes und der entsprechenden Zelle an derselben Position liegen.
- **FitRow** ist nur für Bilder möglich (und nur für Excel) und setzt die Höhe der Reihe/Zelle in der sich das Bild befindet auf die Höhe des Bildes. Beispiel: `|FitRow:Yes`. Je nach Systemeinstellung der Skalierung kann es zu Problemen mit der korrekten Berechnung der Höhe kommen, darum bitte prüfen ob die 'benutzerdefinierte Skalierung' von Windows auf 100% eingestellt ist.
- Wird **OldValue** angegeben, wird der Wert des vorhergehenden Input-Datensatzes ausgegeben. Handelt es sich um den aller ersten Datensatz, wird der Platzhalter leer gelassen.
- **NonZero:** Bei Attributen vom Typ Zahl oder String kann angegeben werden, was ausgegeben werden soll, falls der Wert nicht 0 oder der String nicht leer ist (z.B. Zeilenumbruch). Mit `NonZero:nicht Null` wird z.B. anstelle einer Zahl der Text 'nicht Null' ausgegeben, ausser falls der Wert des Platzhalters ist 0. Soll die Zeile ganz unterdrückt werden, kann der RemoveRow-Platzhalter (Platzhalter, nicht Tag) verwendet werden.
- **Zero:** Bei Attributen vom Typ Zahl oder String kann angegeben werden, was ausgegeben werden soll, falls der Wert 0 oder der String leer ist. Soll die Zahl 0 nicht angezeigt werden, kann diese mit diesem Attribut auch unterdrückt werden: `{CO.CashDiscount|Zero:}`. Soll die Zeile ganz unterdrückt werden, kann der RemoveRow-Platzhalter (Platzhalter, nicht Tag) verwendet werden.
- **True:** Bei Attributen vom Typ Boolean/String kann angegeben werden, was ausgegeben werden soll, falls der Wert 'True' ist, z.B. `True:ausgewählt`. Soll die Zeile ganz unterdrückt werden, kann der RemoveRow-Platzhalter (Platzhalter, nicht Tag) verwendet werden.
- **False:** Bei Attributen vom Typ Boolean/String kann angegeben werden, was ausgegeben werden soll, falls der Wert 'False' ist, z.B. `False:abgemeldet`. Soll die Zeile ganz unterdrückt werden, kann der RemoveRow-Platzhalter (Platzhalter, nicht Tag) verwendet werden.
- Beispiel 1: `{CI.InvoiceDate|Format:dd.MM.yyyy}`
- Beispiel 2: `{CO.CashDiscount|Format:"Skonto: "##,##0.00"% "|Zero:}`
- Beispiel 3: `{P.VornamePartner|ColorCondition:{P.NamePartner}="Posity AG";255,50,0}`
- Beispiel 4: `{Rechnung.Betrag|ColorCondition:{Rechnung.Betrag}<0.0;255,0,0}`

Background Color Cell ::=

```
"{BGC:"<RangeValue>{" | "<RangeValue>"}"; "<AttributeWithNr>"; "<ColorR>", "<ColorG>", "<ColorB>{" | "<ColorR>", "<ColorG>", "<ColorB>"}" }
```

- Legt die Farbe eines Zellenhintergrunds fest. Falls der Wert (Value) innerhalb angegebener Grenzen liegt ( $\text{Value} \geq \text{RangeValue}(x)$  und  $\text{Value} < \text{RangeValue}(x+1)$ ) wird der Zellenhintergrund entsprechend der angegebenen Farbe(x) eingefärbt.
- Position des Platzhalters: In der einzufärbenden Zelle
- Beispiel 1: {BGC:0|2;1.2;0,200,120}
- Beispiel 2: {BGC:0|2|99;{COI.Quantity};0,200,120|0,10,10}

BorderCell ::=

```
"{BOC:"<AttributeWithNr>{" | "<AttributeWithNr>"}"; ("Line" | "Dot" | "Dash" | "0" | "1" | "2")  
"; "  
("Thin" | "Medium" | "Thick" | "0" | "1" | "2") }"
```

- Formatiert den umgebenden Rahmen der Zellen in der aktuellen Zeile an den entsprechenden Spaltenpositionen (AttributeWithNr) gemäss den Parametern. Wird keine AttributeWithNr angegeben (leere Liste), oder ist die AttributeWithNr 0 wird kein Rahmen gesetzt.  
ACHTUNG: die Tabelle darf keinerlei verbundene Zellen aufweisen!
- Position des Platzhalters: In der einzufärbenden Zeile (kann in der ersten Spalte stehen, aber den Rahmen der Zelle in der dritten Spalte formatieren).
- Die Werte 0, 1 oder 2 werden entsprechend 'übersetzt' (Line = 0, Dot = 1, etc.)
- Beispiel 1: {BOC:1;Line;Thin} oder auch {BOC:1;0;0}
- Beispiel 2: {BOC:1|3|5;Dot;Medium}
- Beispiel 3: {BOC:{CO.Number};Line;Thin}

DecimalSeparator ::= "{ DecimalSeparator: "<AnyString>" }

- Hiermit kann angegeben werden, welches Dezimaltrennzeichen in den Formatangaben (FormatPattern) verwendet werden soll (in der Formatangabe wird das Dezimaltrennzeichen immer als Punkt angegeben). Diese Einstellung wird unabhängig vom Betriebssystem und allfälliger Ländereinstellungen verwendet.
- Position des Platzhalters: An beliebiger Stelle im Dokument (ist optional)
- Beispiel: {DecimalSeparator:;} das Komma wird als Dezimaltrennzeichen verwendet.
- Kann nicht in Excel-Vorlagen verwendet werden.

```
DocumentBreak ::= "{ DocumentBreak: "("<TableName>|<TableAlias>")  
{" | Dependence: "<Attribute>", "<Attribute>"  
" | PageBreak: "("Yes" | "No") }"
```

- Mittels diesem Platzhalter kann ein Seriendruck-ähnliches Layout erzeugt werden. An erster Stelle wird die Master-Input-Tabelle (z.B. Offerte) angegeben, für welche pro Record ein 'Document' aufgrund der Vorlage erzeugt werden soll. Wurde ein Alias-Name vergeben, muss dieser Alias-Name verwendet werden.
  - Im Word wird nach der Verarbeitung jedes Records der Master-Input-Tabelle ein Seitenumbruch (der Seitenumbruch selbst kann unterdrückt werden) eingeführt und anschliessend mit dem nächsten Datensatz wieder eine Word-Vorlage verarbeitet. Sind z.B. 100 Offerten in der Master-Input-Tabelle, wird die Word-Vorlage 100 Mal angewendet (das Resultat ist ein einzelnes Word-Dokument, in welchem 100 Mal die Vorlage verarbeitet wurde).
  - Im Excel wird ein neues Sheet eingefügt, falls der Seitenumbruch nicht unterdrückt ist, ansonsten wird die nächste zu verarbeitende Vorlage an die letzte verwendete Zeile des aktuellen Sheets angefügt.
- Sollen in einer Vorlage Abhängigkeiten zwischen Input-Tabellen berücksichtigt werden (sollen z.B. nur die Offertpositionen der aktuell verarbeiteten Offerte berücksichtigt werden), so kann dies im Document-Break-Platzhalter mit dem Dependence-Attribut angegeben werden (siehe Beispiel unten). Es können dabei beliebig viele Abhängigkeiten (als Primärschlüssel-Fremdschlüssel-Tupel) zur Master-Input-Tabelle angegeben werden.

- Wird **PageBreak** auf Yes gesetzt, wird jedes neue Tupel der Master-Input-Tabelle im Word auf einer neuen Seite ausgegeben, respektive im Excel auf einem neuen Sheet eingefügt, ansonsten sind die Daten ohne Seitenumbruch im selben Dokument, respektive auf demselben Sheet hintereinander.
- Beispiel:  
{DocumentBreak:CustomerOffer|Dependence:CustomerOffer.PK\_CustomerOffer,CustomerOfferItem.FK\_CustomerOffer|PageBreak:No}
- Position des Platzhalters: An beliebiger Stelle im Dokument (ist optional)

```
ExportFile ::= "{Exportfile:"<AbsolutePathFileAndExtension>
["|Open:" ("No"|"Yes") ]
}"
```

- Gibt an, dass das erzeugte Dokument in einer Datei gespeichert werden soll. Erlaubte Formate der Zieldatei sind: docx, doc, txt, rtf, pdf, xps und xml. Je nach Format muss mit Darstellungsverlusten gerechnet werden.
- Dieser Platzhalter hat KEINE Wirkung auf den Dateinamen der erzeugten Datei, falls beim Aufruf der Generierung mittels Parameter der Name der Export-Datei übergeben wird! Wird weder mittels Parameter, noch mittels Exportfile-Platzhalter ein Dateiname angegeben, erhält die erzeugte Datei einen Defaultnamen, erzeugt aus dem Namen der Vorlage plus einem Datum-Zeit-Stempel (z.B. "ProtectionLevel\_20151223170849737.docx"). Soll der Defaultname verwendet werden, kann anstelle des Filenamens "Default" angegeben werden.
- Im Tag AbsolutePathFileAndExtension erfolgt die Angabe des absoluten Pfades, plus Filename, plus Fileextension des Importfiles. Der Name muss gemäss Windows-Spezifikation angegeben werden, z.B. "C:\Temp\RechnungDaten.txt".
- Wird **"Open:Yes"** angegeben, wird die erstellt Datei automatisch geöffnet.
- Der Exportfile-Platzhalter wird als letzter Schritt in der Verarbeitung interpretiert. Daher ist es möglich, im Filenamens Platzhalter zu verwenden (siehe 2. Beispiel)!
- Beispiel 1: {Exportfile:C:\Temp\Offerte.docx|Open:No}
- Beispiel 2: {Exportfile:C:\Temp\Offerte-{Cl.OfferNumber}.pdf|Open:Yes}
- Beispiel 3: {Exportfile:Default|Open:Yes}
- Position des Platzhalters: An beliebiger Stelle im Dokument (ist optional)

```
FooterBreak ::=
```

```
"{FooterBreak:"(<TableName>|<Attribute>){", "(<TableName>|<Attribute>)}"}"
```

Schlusszeile zu einen Bereich identischer Werte in einer Liste. Nach einem Wertewechsel des angegebenen Attributes, der Attributliste oder dem Ende der angegebenen Input-Tabelle (der Attributname ist optional) wird diese Zeile ans Ende der Liste kopiert und anschliessend werden die Attributwerte der kopierten Zeile ersetzt. Muss in einer Word-, Excel-Liste in der ersten Spalte stehen.

- Beispiel: {FooterBreak:COI.NameProductGroup}
- Position des Platzhalters: In der ersten Spalte einer Liste. Die Zeile enthält die Daten die beim Umbruch angezeigt werden sollen.
- Bitte beachten: wenn Diagramme (Bar Chart, Pie Chart) verwendet werden funktioniert die Anpassung der Datenbereiche der Charts nicht, da FooterBreaks diese unterbrechen. Ein möglicher Workaround ist es die im Chart verwendeten Daten auf einem separaten Arbeitsblatt einzufügen und den Datenbereich der Charts darauf zu beziehen.

```
FormatParagraphIndentation ::= "{FPI:Left:"<Number>","Right:"<Number>","
("FirstLine:"<Number>|"Hanging:"<Number>)
["|Bullet:"
(<SymbolUnicode>["|<Level>"], "<Fontname>|
<SymbolShortcut>["|<Level>"])]}"
```

- Formatiert die Einzüge eines Word-Paragraphen. Kann nicht in Excel-Vorlagen verwendet werden. Darf nur ein Mal pro Paragraph verwendet werden.
- Wird in der Regel zusammen mit dem Zero- oder NonZero-Tag verwendet. Darf aber auch in den importierten Daten enthalten sein (= im Text eines Attributes).



- Die Zahlen geben die Einzüge in Zentimeter (cm) an. Die Angaben erfolgen in der Reihenfolge: Links (Left, linker Abstand zum Dokumentrand), Rechts (Right, rechter Abstand zum Dokumentrand), erste Zeile (FirstLine), resp. Einzug nach der ersten Zeile (Hanging). FirstLine und Hanging kann nicht gleichzeitig angegeben werden.
- Bullet (Optional): Soll der Paragraph ein Aufzählungszeichen (respektive eine Aufzählung) erhalten, kann dies durch das Schlüsselwort Bullet aktiviert werden. In Word (und im generierten Dokument) basieren Aufzählungen auf Aufzählungsvorlagen welche die Dokumentaufzählungszeichen auf jedem Aufzählungslevel, Masse für die Einrückung und weitere Formatierungsdaten enthalten. Der Bullet-Parameter setzt daher voraus, dass eine Aufzählungsvorlage mit dem gewünschten Aufzählungszeichen im Word-Dokument, in welchem das FormatParagraphIndentation-Tag mit Bullet-Option verwendet wird, enthalten ist.

Eine Aufzählungsvorlage mit gewünschtem Aufzählungszeichen ist im Word enthalten, wenn diese im Dropdown-Menü 'Aufzählungszeichen > Dokumentaufzählungszeichen' sichtbar ist. Eine Aufzählungsvorlage wird im Dokument sichtbar, wenn diese in einer Aufzählung verwendet wird (das Dokumentaufzählungszeichen ist auch noch vorhanden, wenn die Aufzählung im Text wieder entfernt wird) oder über dasselbe Dropdown-Menü 'Aufzählungszeichen > Neues Aufzählungszeichen definieren...' können neue Aufzählungsvorlagen erstellt werden, die sodann als Dokumentaufzählungszeichen im bearbeiteten Word-Dokument verwendet werden können. Mit dem Bullet-Parameter wird über das Symbol und den Font das zu verwendende Dokumentenaufzählungszeichen referenziert. Es wird das Dokumentenaufzählungszeichen verwendet, bei dem das Symbol und der Font mit der Angabe im Bullet Parameter übereinstimmen. Wenn identische Dokumentaufzählungszeichen vorkommen, wird das Erste in der Liste verwendet. Dies bedeutet, es wird immer die Dokumentaufzählungszeichen-Ausrichtung (Links, Rechts oder Zentriert) vom ersten gefundenen Dokumentaufzählungszeichen verwendet.

- Symbol-Unicode: Jedes Symbol aus der Symbolbibliothek (von Word) kann zum Referenzieren eines Dokumentaufzählungszeichens verwendet werden (heisst jedes Symbol kann als Aufzählungssymbol verwendet werden), indem im Tag dessen Unicode-Wert als Dezimal-Wert angegeben wird. In der Symbolbibliothek ist der Hex-Wert eines Symbolen ersichtlich. Beispiel: ☞ 0F12 (Hex) → Taschenrechner App → 3858 (Dezimal), Font-Name: Microsoft Himalaya: {FPI:Left:0,Right:0,Hanging:0,Bullet:3858,0,Microsoft Himalaya}
- Symbol-Shortcut: Folgende Symbol-Shortcuts können angewendet werden (fest vorgegeben), vorausgesetzt die Aufzählungsvorlage ist im Word-Dokument vorhanden!
  - Wird erstellt mit {FPI:Left:0,Right:0,Hanging:0,Bullet:o}
  - Wird erstellt mit {FPI:Left:0,Right:0,Hanging:0,Bullet:O}
  - Wird erstellt mit {FPI:Left:0,Right:0,Hanging:0,Bullet:oo}
  - Wird erstellt mit {FPI:Left:0,Right:0,Hanging:0,Bullet:OO}
  - Wird erstellt mit {FPI:Left:0,Right:0,Hanging:0,Bullet:>}
  - ❖ Wird erstellt mit {FPI:Left:0,Right:0,Hanging:0,Bullet:<>}
  - Wird erstellt mit {FPI:Left:0,Right:0,Hanging:0,Bullet:[]}
- Level (optional): Definiert den Einrückungslevel (voreingestellte Absatz-Einstellungen) der Aufzählung. Zahlenwert 0-8, 0 keine Einrückung (ist Default), Beispiel Bullet mit Einrückungslevel 1: {FPI:Left:0,Right:0,Hanging:0,Bullet:>,1}
- Font (muss angegeben werden, wenn ein Unicode-Symbol verwendet wird): Namen vom Font, der das verwendete Symbol enthält. Beispiel: {FPI:Left:0,Right:0,Hanging:0,Bullet:167,0,Walbaum Display Heavy}
- Bullet + Left,Right,Hanging,Firstline: Haben die Indentation Left,Right,Hanging,Firstline Parameter Werte > 0 und werden diese mit dem Parameter Bullet verwendet, überschreiben diese die Angaben in der Aufzählungsvorlage resp. des Dokumentaufzählungszeichens, wo diese auch enthalten sind! Beispiel Aufzählung ist 1cm vom linken Rand ausgerichtet: {FPI:Left:1.0,Right:2.5,FirstLine:0.0,Bullet:oo}
- Beispiel (no bullet): {FPI:Left:1.0,Right:2.5,FirstLine:0.0}
- Position des Platzhalters: An beliebiger Stelle innerhalb des betroffenen Paragraphen.



HeaderBreak ::=

"{HeaderBreak: "<TableName>|<Attribute>"}", "<TableName>|<Attribute>"}"

- Überschrift über einen Bereich identischer Werte in einer Liste. Nach einem Wertewechsel des angegebenen Attributes, der Attributliste oder dem Ende der angegebenen Input-Tabelle (der Attributname ist optional) wird die Zeile ans Ende der Liste kopiert und anschliessend werden die Attributwerte der kopierten Zeile ersetzt. Muss in einer Word-, Excel-Liste in der ersten Spalte stehen.
- Beispiel: {HeaderBreak:COI.NameProductGroup}
- Position des Platzhalters: In der ersten Spalte einer Liste. Die Zeile enthält die Daten die beim Umbruch angezeigt werden sollen.
- Bitte beachten: wenn Diagramme (Bar Chart, Pie Chart) verwendet werden funktioniert die Anpassung der Datenbereiche der Charts nicht, da HeaderBreaks diese unterbrechen. Ein möglicher Workaround ist es die im Chart verwendeten Daten auf einem separaten Arbeitsblatt einzufügen und den Datenbereich der Charts darauf zu beziehen.

IncludeHeaderFooter ::= "{IncludeHeaderFooter}

- Kopf- und Fusszeilen des Dokumentes werden in der Regel aus Performancegründen nicht nach Platzhaltern durchsucht. Falls diese auch durchsucht werden sollen, muss dieser Platzhalter zwingend eingefügt werden.
- Beispiel: {IncludeHeaderFooter}
- Position des Platzhalters: An beliebiger Stelle im Haupttext des Dokumentes, nicht innerhalb der Kopf- oder Fusszeilen des Dokumentes

Importfile ::= "{Importfile:"<AbsolutePathFileAndExtension>"}

- Dieses Tag hat KEINE Wirkung, falls beim Aufruf der Generierung mittels Parameter der Name der Import-Datei übergeben wird!
- In AbsolutePathFileAndExtension erfolgt die Angabe des absoluten Pfades, plus Filename, plus Fileextension des Importfiles. Der Name muss gemäss Windows-Spezifikation angegeben werden, auch für Apple-Betriebssysteme! Z.B. "C:\Temp\RechnungDaten.txt".
- Beispiel: {Importfile:C:\Temp\RechnungDaten.txt}
- Position des Platzhalters: An beliebiger Stelle im Dokument (ist optional für Windows-Systeme, ohne Angabe öffnet Fileopen-Dialog)

NewParagraph ::= "{NewParagraph}"

- Teilt den Word-Paragraphen an der definierten Stelle in zwei format-identische Paragraphen. Kann nicht in Excel-Vorlagen verwendet werden.
- Wird in der Regel zusammen mit dem Zero- oder NonZero-Tag verwendet. Darf aber auch in den importierten Daten enthalten sein. Dieses Tag funktioniert nicht innerhalb von Listen, siehe hierfür {NewLine}.
- Beispiel: {NewParagraph}
- Beispiel: {CI.ListPrice|Format:"Listenpreis: "##,##0.00"{NewParagraph}"|DependOn:CI.Discount}
- Position des Platzhalters: An der Stelle, an der der Paragraph aufgeteilt werden soll.

NewLine ::= "{NewLine}"

- Fügt bei einem Paragraphen an der definierten Stelle einen Linienumbruch ein. Kann nicht in Excel-Vorlagen verwendet werden.
- Wird in der Regel zusammen mit dem Zero- oder NonZero-Tag verwendet. Darf aber auch in den importierten Daten enthalten sein.
- Beispiel: {NewLine}
- Beispiel: {CI.ListPrice|Format:"Listenpreis: "##,##0.00"{NewLine}"|DependOn:CI.Discount}
- Position des Platzhalters: An der Stelle, an der der Linienumbruch eingefügt werden soll.

OptionalDocumentPart ::=

"{ODP: "<Attribute>|<String>|<Number><ComparisonOperator>(<Attribute>|<String>|<Number>)"}

- Kennzeichnet eine Tabelle als optional. Ist die Bedingung nicht erfüllt, wird die gesamte Tabelle aus dem Dokument entfernt. Dadurch können gezielt Teile des Dokumentes ein- respektive ausgeblendet werden.
- Beispiel: {ODP:P.NamePartner="Posity AG"}

- Position des Platzhalters: In einer beliebigen Zelle der Tabelle.

`RemoveRow` ::= "`{RemoveRow}`"

- In bestimmten Situationen soll eine Zeile einer Liste nicht in das Dokument integriert werden, z.B. weil in einem Attribut kein Wert enthalten ist (z.B., wenn der Rabatt der Zeile 0 ist). Dieses Tag wird in aller Regel in Kombination mit dem Zero-, NonZero-, True- oder False-Tag des AttributeValue-Platzhalters verwendet.
- Beispiel: `{CO.CashDiscount|Zero:{RemoveRow}}`

`Row` ::= "`{Row}`"

- Wird in einer Liste das Row-Schlüsselwort gesetzt, so wird für jeden verarbeiteten Datensatz in dieser Liste eine einzelne Zeile erzeugt. Das Erstellen einer neuen Zeile wird für jeden Datensatz höchstens einmal ausgeführt (falls mehrere Attribute aufgeführt werden).
- Beispiel: `{Row}`
- Position des Platzhalters: In der ersten Spalte einer Liste. Die Zeile enthält die Daten die beim jedem Datensatz angezeigt werden sollen.

`Spacer` ::= "`{Spacer}`"

- Innerhalb einer einzelnen Liste können mehrere Zeilen erstellt werden, die durch Posity bearbeitet werden (z.B. vom Typ Row, FooterBreak, HeaderBreak). Damit die Rahmen diese Zeilen unabhängig voneinander formatiert werden können, muss zwischen diesen eine Leerzeile eingefügt werden können. Diese Leerzeile soll im finalen Dokument aber nicht mehr erscheinen. Dies wird erreicht, indem diese Leerzeile in der ersten Spalte mit Spacer markiert wird, diese Zeilen werden nach dem Erzeugen des Dokumentes gelöscht.
- Beispiel: `{Spacer}`
- Position des Platzhalters: In der ersten Spalte der Zeile die gelöscht werden soll.

`Sum` ::=

```
"{Sum:{<Attribut>}}"
["|Format:"["<StringWithBreak>"]<FormatPattern>["<StringWithBreak>"]]
["|SumCondition:"<Condition>]
["|ShowConditionAll:"<Condition>;"<Boolean>"]
["|ShowConditionAny:"<Condition>;"<Boolean>"]
"}"
```

- Der Sum-Befehl muss in einer Liste in Kombination mit dem FooterBreak oder HeaderBreak eingesetzt werden. Im Sum-Platzhalter werden die Werte bis zum FooterBreak, resp. HeaderBreak summiert ausgegeben.
- **Format:** Im `<FormatPattern>` kann das Ausgabeformat gemäss Spezifikation des Format-Befehls von C# gesetzt werden, ergänzt um einen vorhergehenden und nachfolgenden String (z.B. `"Total: '#,##0.00'` oder `'#,##0' \%`). Siehe auch Beschreibung zu `<FormatPattern>`.  
Kann nicht in Excel-Vorlagen verwendet werden.
- **SumCondition:** Ist die Bedingung erfüllt, wird das Wert des Attributes in der Summe berücksichtigt. Im Vergleich können nur Attribute des aktuellen Datensatzes verwendet werden. Beide Attribute müssen vom Typ String (alle Typen ausser Numeric werden als String eingesetzt) oder vom Typ Numeric sein. Ist der Vergleich nicht möglich (falsche Typen, falsches Attribut) wird die Bedingung ignoriert.  
Gross/Kleinschreibung wird bei der Auswertung der Vergleichsoperation ignoriert! Attribute müssen in geschweifte Klammern gesetzt werden.
- **ShowConditionAll:** Die Summe wird nur angezeigt, falls für die gebildete Summe die definierte Bedingung für jeden verrechneten Datensatz erfüllt ist. Bedingungen von Datensätzen, die aufgrund des Flags `SumCondition` nicht zur Summenbildung verwendet werden, werden nicht berücksichtigt. `ShowConditionAll` kann mit `ShowConditionAny` kombiniert werden. Mit dem Bool-Wert nach der `ShowCondition` kann die Anzeige der Summe beim nicht Zutreffen der `ShowCondition` gesteuert werden. Mit `"true"` wird die Zeile, in der sich das Sum-Tag befindet, entfernt. Mit `"false"` wird die Summe nicht angezeigt. Die Zeile wird entfernt wenn bei `ShowConditionAll` oder `ShowConditionAny` der Wert `"true"` vorkommt.
- **ShowConditionAny:** Die Summe wird nur angezeigt, falls für die gebildete Summe die definierte Bedingung für irgendeinen verrechneten Datensatz erfüllt ist. Bedingungen von Datensätzen, die aufgrund des Flags `SumCondition` nicht zur Summenbildung verwendet werden, werden nicht berücksichtigt. `ShowConditionAny`

kann mit ShowConditionAll kombiniert werden. Mit dem Bool-Wert nach der ShowCondition kann die Anzeige der Summe beim nicht Zutreffen der ShowCondition gesteuert werden. Mit "true" wird die Zeile, in der sich das Sum-Tag befindet, entfernt. Mit "false" wird die Summe nicht angezeigt. Die Zeile wird entfernt wenn bei ShowConditionAll oder ShowConditionAny der Wert "true" vorkommt.

- Beispiel: {Sum:{COI.Cost}|Format:#,##0.00}
- Beispiel: {Sum:{Rechnung.Betrag}|SumCondition:{Rechnung.Betrag}>0.0}
- Beispiel: {Sum:{RechnungPosition.Betrag}|ShowConditionAll:{RechnungPosition.Option}="True"}
- Beispiel: {Sum:{RechnungPosition.Betrag}|ShowConditionAny:{RechnungPosition.Option}="True";"true"}
- Position des Platzhalters: An der Stelle an der die Summe angezeigt werden soll (in der Liste).

TabChar ::= "{TabChar}"

- Fügt bei einem Paragraphen an der definierten Stelle ein Tabulator-Zeichen ein. Kann nicht in Excel-Vorlagen verwendet werden.
- Wird in der Regel zusammen mit dem Zero- oder NonZero-Tag verwendet. Darf aber auch in den importierten Daten enthalten sein.
- Beispiel: {TabChar}
- Beispiel: {CI.ListPrice|Format:"Listenpreis{TabChar}"##,##0.00|DependOn:CI.Discount}
- Position des Platzhalters: An der Stelle, an der das Tabulator-Zeichen eingefügt werden soll.

TableAlias ::= "{TableAlias:"<TableAliasName>=<TableName>"}"

- Um die Lesbarkeit und Darstellung zu verbessern, können für jeden Input-Tabellennamen Aliasnamen vergeben werden. Achtung, falls ein Aliasname vergeben wird muss dieser zwingend durchgängig verwendet werden. Wird z.B. Beispiel für die Input-Tabelle CustomerInvoice der Aliasname CI gesetzt (siehe Beispiel unten), dann kann der Platzhalter statt '{CustomerInvoice.InvoiceDate}' mit '{CI.InvoiceDate}' geschrieben werden.
- Beispiel: {TableAlias:CI=CustomerInvoice}
- Position des Platzhalters: An beliebiger Stelle im Dokument (ist optional)

ThousandSeparator ::= "{ThousandSeparator:"<AnyString>"}"

- Hiermit kann angegeben werden, welches Tausendertrennzeichen in den Formatangaben (FormatPattern) verwendet werden soll (in der Formatangabe wird das Tausendertrennzeichen immer als Komma angegeben). Diese Einstellung wird unabhängig vom Betriebssystem und allfälliger Ländereinstellungen verwendet.
- Beispiel: {ThousandSeparator:','}
- Position des Platzhalters: An beliebiger Stelle im Dokument (ist optional)
- Kann nicht in Excel-Vorlagen verwendet werden.

UpdateFields ::= "{UpdateFields}"

- Sollen die Feldfunktionen im Dokument oder in Charts (Bild und Datenbereich) automatisch aktualisiert werden, muss dieser Platzhalter gesetzt werden – ansonsten müssen diese nach dem Öffnen des Dokumentes von Hand aktualisiert werden. Das automatische Aktualisieren funktioniert nur dann, wenn das Dokument danach auch tatsächlich im Word, resp. Excel durch den Verarbeitungsprozess geöffnet wird.
- Achtung: Es werden nur Feldfunktionen im Haupttext (nicht im Header oder Footer) aktualisiert.
- Beispiel: {UpdateFields}
- Position des Platzhalters: An beliebiger Stelle im Dokument (ist optional)

ZebraLines ::=

"{Zebra:"<ColorR>","<ColorG>","<ColorB>";"<ColorR>","<ColorG>","<ColorB>"}"

- In den Listen können die Zeilen mit abwechselnden Farben hinterlegt werden. Die erste Farbe wird jeweils für die erste Zeile verwendet, danach wird mit der zweiten Farbe jede Zeile abwechselnd eingefärbt. Sind zwischen den Zeilen andere Zeilen eingebettet (z.B. Header oder Footer), oder wird in eine neue Liste gewechselt, wird wieder mit der ersten Farbe gestartet.
- Beispiel: {Zebra:190,190,190;220,220,220}

- Position des Platzhalters: In der Zeile welche zebraartig eingefärbt werden soll (in irgendeiner Spalte der Zeile).

## 4. Beispiel Word-Vorlage und erzeugtes Dokument

Auf den folgenden Seiten wird ein Beispiel einer Vorlage und eines damit erzeugten Dokumentes gezeigt. In der Vorlage wurden die Rahmen der Tabellen grün eingefärbt, damit die Vorlage besser interpretiert werden kann. Bei der Generierung wurde diese Färbung nicht verwendet.

Der Input enthält zwei Tabellen CO (CustomerOffer) und COI (CustomerOfferItem).

{CO.FIRSTNAMEPARTNERCUSTOMER}{CO.FirstNamePartnerCustomer|Zero:|NonZero:  
{CO.NAMEPARTNERCUSTOMER}  
{CO.MAILADDRESSLINE1PARTNERCUSTOMER}  
{CO.MAILADDRESSLINE1NUMBERPARTNERCUSTOMER}  
{CO.POSTALCODECITYPARTNER} {CO.NAMEORTPARTNER}

**ANGEBOT**  
**{CO.NAMEOFFER}**

Nr. {CO.OfferNumber}  
Winterthur, {CO.OfferDate|Format:dd.MM.yyyy}

*{CO.ImageConstruction|MaxWidth:600|MaxHeight:300}*

Anlagenstandort: {CO.NamePlantLocation}  
Installation: {CO.NameSetupTyp}  
Neigung: {CO.Inclination|Format:#0}  
Ausrichtung: {CO.Orientation|Format:#0}  
Modulfläche {CO.ModuleArea|Format:#,##0} m<sup>2</sup>  
Installierte Leistung: {CO.InstalledOutput|Format:#,##0.00} kWp  
Spezifischer Ertrag: {CO.SpecificOutput|Format:#,##0} kWh/kWp  
Geschätzter Jahresertrag: {CO.EstimatedAnualOutput|Format:#,##0} kWh  
Inbetriebnahme bis: {CO.CommissioningDate|Format:dd.MM.yyyy}  
Projektkosten inkl. MwSt.: {CO.ProjectCostsWithTax\_Currency}  
{CO.ProjectCostsWithTax|Format:#,##0.00}; {CO.CostsPerkWp\_Currency}  
{CO.CostsPerkWp|Format:#,##0}/kWp





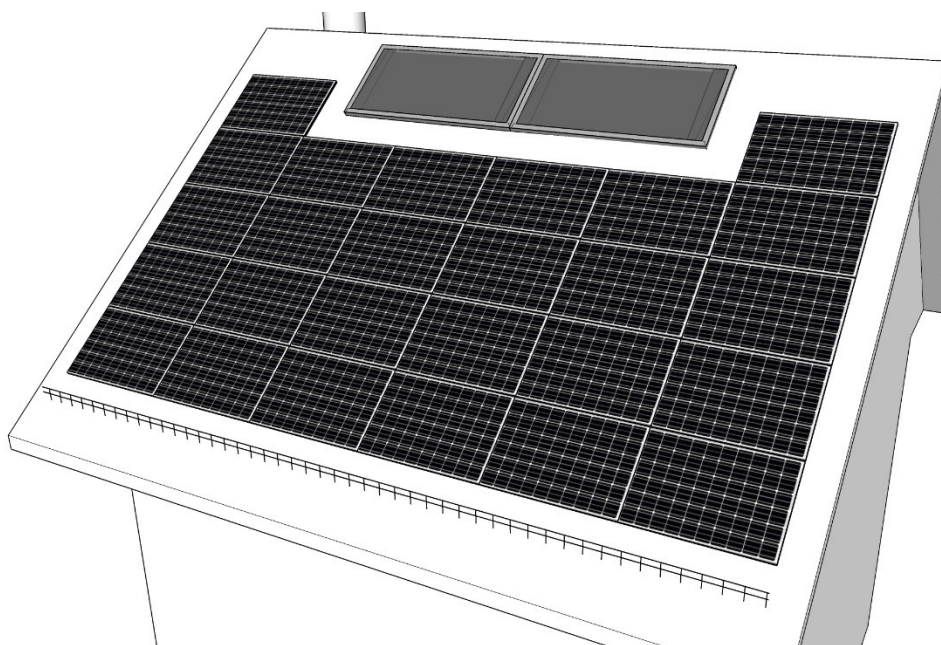
Angebotsgültigkeit:	{CO.ValidityOfOffer} Tage
Zahlungskonditionen:	{CO.TermsOfPayment}{CO.CashDiscount Format:" Skonto: "##,##0.00"% " Zero:} {CO.PaymentPeriodCashDiscount Format:"("##,##0" Tage)" Zero:}
Zahlungsfrist:	{CO.PaymentPeriod Format:#,##0} Tage
Garantien:	{CO.Warranty}
Auftragsbestätigung:	{CO.TextFeederOffer}
Ort und Datum	Unterschrift, die Bauherrschaft

{TableAlias:CO=CustomerOffer}{TableAlias:COI=CustomerOfferItem}{ThousandSeparator:'}{DecimalSeparator:.  
.}

FIRMENNAME  
STRASSE  
8400 WINTERTHUR

**ANGEBOT**  
**AUFDACHANLAGE FAMILIE XYZ**

Nr. A150000  
Winterthur, 27.11.2015



Anlagenstandort:	Schrägdach
Installation:	Angebaut
Neigung:	40°
Ausrichtung:	Süden
Modulfläche	42 m <sup>2</sup>
Installierte Leistung:	8.58 kWp
Spezifischer Ertrag:	1'040 kWh/kWp
Geschätzter Jahresertrag:	8'923 kWh
Inbetriebnahme bis:	30.03.2016
Projektkosten inkl. MwSt.:	CHF 28'284.30; CHF 3'297/kWp

Menge	Beschreibung	Einzelpreis	CHF/kWp	Preis
26 Stk.	PV-Modul BenQ SunForte 330 Wp Hochleistungs-Qualitätsmodul Modulwirkungsgrad 20.3% (LxBxH): 1559 x 1046 x 46 mm Gewicht: 18.6 kg Pmax 330 W (-0/+3 Leistungstoleranz) Umpp 54.7 V, Imp 6.04 A IEC 61215, IEC 61730	404.35	1'225	10'513.15
26 Stk.	Montagesystem mit Einlegeprofilen, schwarz Einlegesystem im Kreuzverbund von Creotecc schwarz eloxiert mit Dachhaken für Pfannenziegel	84.50	256	2'197.00
1 Stk.	Wechselrichter Fronius Symo 8.2-3-M AC-Ausgangsleistung 8'200 VA 3 phasig max. 13.5A Wirkungsgrad 97.8% Schutzklasse IP 55 2 MPP Tracker 2+2 DC Anschlüsse / Gewicht 21.9 kg	2'034.50	237	2'034.50
1 Stk.	Fronius Smart Meter Messung und Visualisierung des Eigenverbrauchs bidirektionaler Zähler 63A/3ph Ethernetanschluss bauseits	392.08		0.00 (Option)
1 Stk.	DC Ü-Schutz PV DC 2INx2 2MPPT 2SPD WM4 1000 V GAK für Wechselrichter mit 2 MPPT Eingänge 2+2, Ausgänge 1+1, Anschluss über Stecker WM4, Überspannungsschutz Typ II, HxBxT: 360 x 360 x 171 mm	333.59	39	333.59
120 m	Solarkabel 6 mm <sup>2</sup> halogenfrei, abriebfest, diverse Farben +/-	0.94	13	112.32
10 m	Alurohr M40 für Steigzone, inkl. Rohrschellen und Bogen	7.80	9	78.00
15 m	Potentialausgleich 16 mm <sup>2</sup>	1.22	2	18.23
8 Stk.	Solarstecker MC4 1 Paar, Stecker und Buchse	3.12	3	24.96
<b>Material</b>				<b>15'311.75</b>
35 h	Montage Unterkonstruktion und Module	85.01	347	2'975.18
25 h	DC Elektromontage	92.00	268	2'300.00
1 Stk.	AC Elektromontage vom Wechselrichter bis zum Hausanschlusskasten Sicherungen, Überspannungsschutz, Trennschalter Inkl. Installationsanzeige, Sicherheitsnachweis	1'500.00	175	1'500.00
<b>Arbeit / Montage</b>				<b>6'775.18</b>

Menge	Beschreibung	Einzelpreis	CHF/kWp	Preis
15 h	Planung & Dokumentation - Meldeformular Solaranlagen - Anmeldung Einmalvergütung oder andere Fördermittel - EEA Anmeldung bei lokalen EW - Auditbegleitung und WR Registrierung (inkl. Gebühren)	104.00	182	1'560.00
4 h	Transporte Materiallieferung und Entsorgung	120.00	56	480.00
<b>Dienstleistung / Planung</b>				<b>2'040.00</b>
1 Stk.	Sicherheitsgerüst SUVA konform	1'900.00	221	1'900.00
<b>Partnerfirmen</b>				<b>1'900.00</b>
4 Stk.	ABS-Loop für Montage und späteren Wartungszugang	40.56	19	162.24
<b>Sicherheit</b>				<b>162.24</b>
Netto				26'189.16
8% MWSt.				2'095.13
<b>Werkpreis</b>				<b>CHF 28'284.30</b>

Angebotsgültigkeit: 60 Tage

Zahlungskonditionen: 70% bei Erhalt aller Bewilligungen  
30% nach Inbetriebnahme

Zahlungsfrist: 14 Tage

Garantien: Arbeit Solarville 2 Jahre  
Produkte gemäss Datenblätter der Hersteller  
(Gewährleistung Solarville 2 Jahre)

Auftragsbestätigung: Mit den allgemeinen Geschäftsbedingungen, welche unter [www.solarville.ch/agb](http://www.solarville.ch/agb) einsehbar sind, bin ich einverstanden. Ich bestätige die Offerte mit Beilagen komplett verstanden zu haben und erteile der Solarville AG den Auftrag, ihre oben genannten Leistungen auszuführen.

---

Ort und Datum

Unterschrift, die Bauherrschaft

## 5. Beispiel Excel-Vorlage und erzeugtes Dokument

Auf den folgenden Seiten wird ein Beispiel einer Vorlage und eines damit erzeugten Dokumentes gezeigt.

Der Input enthält zwei Tabellen CO (CustomerOffer) und COI (CustomerOfferItem).

{Importfile:C:\temp\OfferteDaten.txt}{TableAlias:CO=CustomerOffer}{TableAlias:COI=CustomerOfferItem}{Exportfile:C:\temp\output\solarville\_offerte.xlsx|Open:Yes}

Kundendaten:

{CO.FirstNamePartnerCustomer} {CO.NamePartnerCustomer}  
{CO.MailAddressLine1PartnerCustomer} {CO.MailAddressLine1NumberPartnerCustomer}  
{CO.PostalCodeCityPartner} {CO.NameOrtPartner}

Offertendaten:

Nr. {CO.OfferNumber}  
Angebot: {CO.NameOffer}  
Datum: {CO.OfferDate|Format:dd.MM.yyyy}

Standort: {CO.NamePlantLocation}  
Installation: {CO.NameSetupTyp}  
Neigung: {CO.Inclination|Format:#0}  
Ausrichtung: {CO.Orientation|Format:#0}  
Modulfläche: {CO.ModuleArea|Format:#,##0} m²  
Installierte Leistung: {CO.InstalledOutput|Format:#,##0.00} kWp  
Spezifischer Ertrag: {CO.SpecificOutput|Format:#,##0} kWh/kWp  
Geschätzter Jahresertrag: {CO.EstimatedAnualOutput|Format:#,##0} kWh  
Inbetriebnahme bis: {CO.CommissioningDate|Format:dd.MM.yyyy}  
Projektkosten inkl. MwSt.: {CO.ProjectCostsWithTax\_Currency} {CO.ProjectCostsWithTax|Format:#,##0.00};  
{CO.CostsPerkWp\_Currency} {CO.CostsPerkWp|Format:#,##0}/kWp

{CO.ImageConstruction|MaxWidth:600|MaxHeight:300|FitRow:Yes}

Offerte Positionen:				
Menge	Beschreibung	Einzelpreis	CHF/kWp	Preis



<div>{Row}{COI.Quantity Format:#,##0 Zero:} {COI.ShortNameMassUnit}{Zebra:190,190,190;220,220,220}</div> <div>{Spacer}</div> <div>{FooterBreak:COI.NameProductGroup}{COI.NameProductGroup OldValue}{BOC:5;0;0}{BGC:0 2;1;201,255,233}</div>	<div>{COI.NameProduct}{COI.DescriptionProduct} {COI.NameProductGroup}</div> <div>{BGC:0 2;1;201,255,233}</div>	<div><div><div>{COI.RetailPricePerProduct Zero:}</div><div>{COI.Option True: {COI.Cost Format:#,##0.00} (Option) False:{COI.Cost}} {COI.ByClient True:(bauseits)  False:}</div></div><div><div>{COI.CostPerOutput Zero:}</div><div>{COI.CostPerOutput Zero:}</div></div><div><div>{COI.CostPerOutput Zero:}</div><div>{COI.CostPerOutput Zero:}</div></div></div> <div><div>{BGC:0 2;1;201,255,233}</div><div>{BGC:0 2;1;201,255,233}</div><div>{Sum:COI.Cost}{BGC:0 2;1;201,255,233}</div></div>
<div>Netto:</div> <div>8% MWSt.:</div> <div>Werkpreis:</div>		<div>{CO.ProjectCosts}</div> <div>{CO.MWSt}</div> <div>{CO.ProjectCostsWithTax_Currency} {CO.ProjectCostsWithTax Format:#,##0.00}{BOC:5;0;0}</div>

Angebotsgültigkeit:

{CO.ValidityOfOffer} Tage

Zahlungskonditionen:

{CO.TermsOfPayment}

{CO.CashDiscount|Format:  
"Skonto: \"##,##0.00\"%  
\"|Zero:}

{CO.PaymentPeriodCashDiscount|Format:(\"##,##0\"  
Tage)\"|Zero:}

Zahlungsfrist:

{CO.PaymentPeriod|Format:#,##0} Tage

**Garantien:**

{CO.Warranty}

**Auftragsbestätigung:**

{CO.TextFeederOffer}

**Ort und Datum**

{UpdateFields}

{IncludeHeaderFooter}

**Unterschrift, die  
Bauherrschaft**

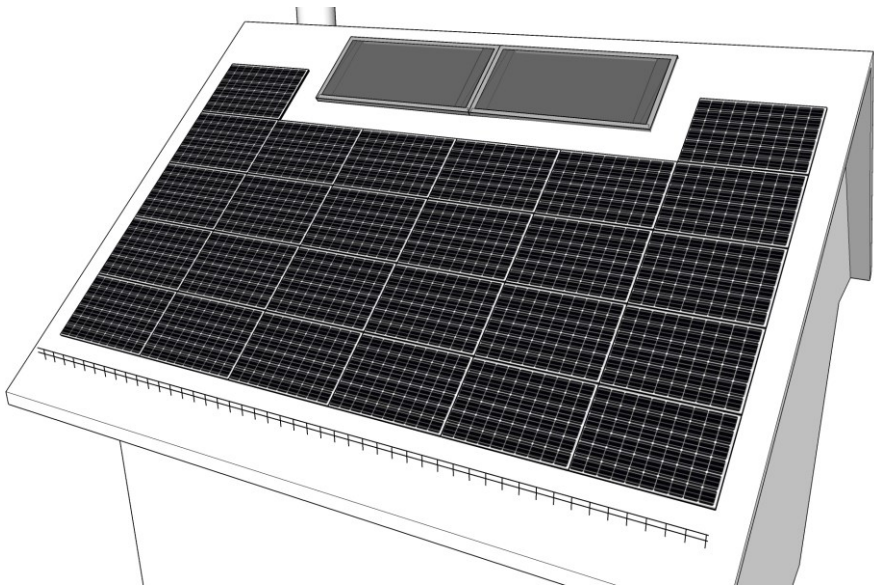
**Kundendaten:**

FAMILIENNAME  
STRASSE  
8400 WINTERTHUR

**Offertendaten:**

Nr. A150000  
Angebot: Aufdachanlage Familie Nagel  
Datum: 27.11.2015

<b>Standort:</b>	Schrägdach
<b>Installation:</b>	Angebaut
<b>Neigung:</b>	40°
<b>Ausrichtung:</b>	Süden
<b>Modulfläche:</b>	42 m2
<b>Installierte Leistung:</b>	8.58 kWp
<b>Spezifischer Ertrag:</b>	1'040 kWh/kWp
<b>Geschätzter Jahresertrag:</b>	8'923 kWh
<b>Inbetriebnahme bis:</b>	30.03.2016
<b>Projektkosten inkl. MwSt.:</b>	CHF 28'284.30; CHF 3'297/kWp



Offerte Positionen:		
Menge	Beschreibung	Einzelpreis CHF/kWp    Preis
26 Stk.	PV-Modul BenQ SunForte 330 Wp Hochleistungs- Qualitätsmodul Modulwirkungsgrad 20.3% (LxBxH): 1559 x 1046 x 46 mm Gewicht: 18.6 kg Pmax 330 W (-0/+3 Leistungstoleranz) Umpp 54.7 V, Impp 6.04 A IEC 61215, IEC 61730 Material	404.35    1'225.31    10'513.15

26 Stk.	Montagesystem mit Einlegeprofilen, schwarzEinlegesystem im Kreuzverbund von Creotecc schwarz eloxiert mit Dachhaken für Pfannenziegel Material	84.50	256.06	2'197.00
1 Stk.	Wechselrichter Fronius Symo 8.2-3-MAC-Ausgangsleistung 8'200 VA 3 phasig max. 13.5A Wirkungsgrad 97.8% Schutzklasse IP 55 2 MPP Tracker 2+2 DC Anschlüsse / Gewicht 21.9 kg Material	2'034.50	237.12	2'034.50
1 Stk.	Fronius Smart MeterMessung und Visualisierung des Eigenverbrauchs bidirektionaler Zähler 63A/3ph Ethernetanschluss bauseits Material	392.08		0.00 (Option)

1 Stk.	DC Ü-Schutz PV DC 2INx2 2MPPT 2SPD WM4 1000 VGAK für Wechselrichter mit 2 MPPT Eingänge 2+2, Ausgänge 1+1, Anschluss über Stecker WM4, Überspannungsschutz Typ II, HxBxT: 360 x 360 x 171 mm Material	333.59	38.88	333.59
120 m	Solarkabel 6 mm <sup>2</sup> halogenfrei, abriebfest, diverse Farben +/- Material	0.94	13.09	112.32
10 m	Alurohr M40für Steigzone, inkl. Rohrschellen und Bogen Material	7.80	9.09	78.00
15 m	Potentialausgleich 16 mm <sup>2</sup> Material	1.22	2.12	18.23
8 Stk.	Solarstecker MC41 Paar, Stecker und Buchse Material	3.12	2.91	24.96
<b>Material</b>				<b>15311.75</b>
35 h	Montage Unterkonstruktion und Module Arbeit / Montage	85.01	346.76	2'975.18
25 h	DC Elektromontage Arbeit / Montage	92.00	268.07	2'300.00

1 Stk.	AC Elektromontagevom Wechselrichter bis zum Hausanschlusskasten Sicherungen, Überspannungsschutz, Trennschalter Inkl. Installationsanzeige, Sicherheitsnachweis Arbeit / Montage	1'500.00	174.83	1'500.00
<b>Arbeit / Montage</b>				<b>6775.18</b>
15 h	Planung & Dokumentation-Meldeformular Solaranlagen - Anmeldung Einmalvergütung oder andere Fördermittel - EEA Anmeldung bei lokalen EW - Auditbegleitung und WR Registrierung (inkl. Gebühren) Dienstleistung / Planung	104.00	181.82	1'560.00
4 h	Transporte Materiallieferung und Entsorgung Dienstleistung / Planung	120.00	55.94	480.00
<b>Dienstleistung / Planung</b>				<b>2040.00</b>
1 Stk.	SicherheitsgerüstSUVA konform Partnerfirmen	1'900.00	221.45	1'900.00
<b>Partnerfirmen</b>				<b>1900.00</b>



4 Stk.	ABS-Loopfür Montage und späteren Wartungszugang Sicherheit	40.56	18.91	162.24
<b>Sicherheit</b>				<b>162.24</b>
Netto:				26'189.16
8% MWSt.:				2'095.13
<b>Werkpreis:</b>				<b>CHF 28'284.30</b>

- Angebotsgültigkeit:** 60 Tage
- Zahlungskonditionen:** 70% bei Erhalt aller  
Bewilligungen  
30% nach Inbetriebnahme
- Zahlungsfrist:** 14 Tage
- Garantien:** Arbeit Solarville 2 Jahre  
Produkte gemäss Datenblätter  
der Hersteller  
(Gewährleistung Solarville 2  
Jahre)

**Auftragsbestätigung:**

Mit den allgemeinen  
Geschäftsbedingungen,  
welche unter  
[www.solarville.ch/agb](http://www.solarville.ch/agb)  
einsehbar sind, bin ich  
einverstanden. Ich bestätige  
die Offerte mit Beilagen  
komplett verstanden zu haben  
und erteile der Solarville AG  
den Auftrag, ihre oben  
genannten Leistungen  
auszuführen.

**Ort und Datum**

**Unterschrift, die  
Bauherrschaft**