

# Where to place styles

A real example from Cdes: For making a text node shift down by some distance, you can do something like

```
<style:style style:name="P3" style:family="paragraph">
  <style:paragraph-properties fo:margin-top="5.000cm"/>
</style:style>
(...)
<text:p text:style-name="P3">Datum der Berichtserstellung: 2017-07-29</text:p>
```

The important question is: Where to place that style? The usual way to find this out is, create a small test document, and save it as a fodt file, which is an uncompressed xml. If you do this, the style snippet above is place in office:automatic-styles, i.e.:

```
<office:automatic-styles>
  <style:style style:name="P1" style:family="paragraph" style:parent-style-name="Standard">
    <style:paragraph-properties fo:margin-top="9.999cm" fo:margin-bottom="0cm" loext:contextual-spacing="false"/>
  </style:style>
```

(where parent style, margin-bottom and contextual-spacing are irrelevant, for the latter, I couldn't even find a specification)

So, creating the style under office:automatic-styles in our Java-code seems to be the right thing. But:

- in the generated odt file (which is a zip file), the office:automatic-styles block containing our style is written to styles.xml
- however, in an odt file generated by Libreoffice, the corresponding office:automatic-styles block is written to content.xml

So, the consequence is, that in our generated odt file, the style is not found, and Libreoffice just ignores it silently. And, because in the fodt file you usually use for testing (as it is not zipped, has line breaks, and thus diff is usable sensefully), everything is in one plain file, you cannot trigger the problem there.

The **solution** is simple: If one places the style under

```
<office:styles>
  <style:style style:family="paragraph" style:name="cdes_paragraph9">
    <style:paragraph-properties fo:margin-top="5.000cm"/>
  </style:style>
```

which in our Java world looks like

```
Styles stylesAutomaticStyles = context.getStylesAutomaticStyles();
Styles stylesOfficeStyles = context.getStylesOfficeStyles();
String marginStyle = stylesOfficeStyles.getParagraphStyle(stylesOfficeStyles.
constructParagraphPropertiesWithMargin(null, null, "5.000cm", null));
TextPElement pElement = TextFactory.constructTextNode(officeText, formattedString, marginStyle);
```

(using stylesOfficeStyles instead of stylesAutomaticStyles)

then the style suddenly is used.