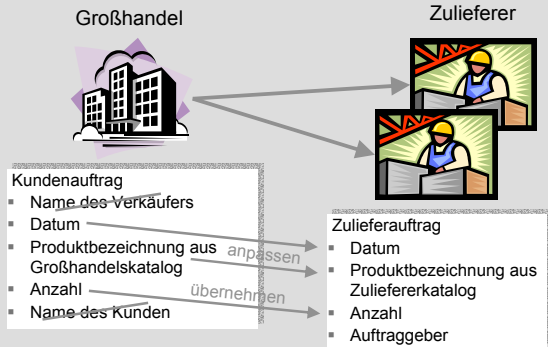


## XML-Inhalt → XML-Inhalt



© Klaus Schild 2003

1

## XML-Inhalt → XML-Inhalt

### Kundenauftrag

```
<?xml version="1.0"?>
<order>
  <salesperson>John Doe</salesperson>
  <item>Production-Class Widget</item>
  <quantity>16</quantity>
  <date>
    <month>1</month>
    <day>13</day>
    <year>2000</year>
  </date>
  <customer>Sally Finkelstein</customer>
</order>
```

XSLT

Zuliefererauftrag

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    <description>Production-Class Widget</description>
    <quantity>16</quantity>
  </item>
</order>
```

© Klaus Schild 2003

2

## Das Stylesheet für das Beispiel

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="xml" indent="yes" encoding="UTF-8"/>
  <xsl:template match="/">
    ...
  </xsl:template>
  <xsl:template match="item">
    ...
  </xsl:template>
</xsl:stylesheet>
```

- method="xml": XML-Deklaration wird erzeugt.
- indent="yes": Prozessor darf (muss aber nicht) Kind-Elemente einrücken.

© Klaus Schild 2003

3

## Datum transformieren

### Kundenauftrag

```
<?xml version="1.0"?>
<order>
  <salesperson>John Doe</salesperson>
  <item>Production-Class Widget</item>
  <quantity>16</quantity>
  <date>
    <month>1</month>
    <day>13</day>
    <year>2000</year>
  </date>
  <customer>Sally Finkelstein</customer>
</order>
```

Zuliefererauftrag

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    <description>Production-Class Widget</description>
    <quantity>16</quantity>
  </item>
</order>
```

© Klaus Schild 2003

4

## Datum transformieren

```
<xsl:template match="/">
  <order>
    <date>
      <xsl:value-of select="order/date/year"/> /
      <xsl:value-of select="order/date/month"/> /
      <xsl:value-of select="order/date/day"/>
    </date>
    ...
  </order>
</xsl:template>
```

Ergebnis

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  ...
</order>
```

© Klaus Schild 2003

5

## Kunden-Element erzeugen

### Kundenauftrag

```
<?xml version="1.0"?>
<order>
  <salesperson>John Doe</salesperson>
  <item>Production-Class Widget</item>
  <quantity>16</quantity>
  <date>
    <month>1</month>
    <day>13</day>
    <year>2000</year>
  </date>
  <customer>Sally Finkelstein</customer>
</order>
```

Zuliefererauftrag

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    <description>Production-Class Widget</description>
    <quantity>16</quantity>
  </item>
</order>
```

© Klaus Schild 2003

6

## Kunden-Element erzeugen

```
<xsl:template match="/">
  <order>
    <date>
      <xsl:value-of select="order/date/year"/>/
      <xsl:value-of select="order/date/month"/>/
      <xsl:value-of select="order/date/day"/>
    </date>
    <customer>Company A</customer>
    ...
  </order>
</xsl:template>
```

Ergebnis

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  ...
</order>
```

## Anzahl übernehmen

### Kundenauftrag

```
<?xml version="1.0"?>
<order>
  <salesperson>John Doe</salesperson>
  <item>Production-Class Widget</item>
  <quantity>16</quantity>
  <date>
    <month>1</month>
    <day>13</day>
    <year>2000</year>
  </date>
  <customer>Sally Finkelstein</customer>
</order>
```

Zuliefererauftrag

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    <description>Production-Class Widget</description>
  </item>
  <quantity>16</quantity>
</order>
```

## Anzahl übernehmen

```
<xsl:template match="/">
  <order>
    <date>...</date>
    <customer>Company A</customer>
    <item>
      ...
      <quantity>
        <xsl:value-of select="order/quantity"/>
      </quantity>
    </item>
  </order>
</xsl:template>
```

Ergebnis

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item> ...
    <quantity>16</quantity>
  </item>
</order>
```

## Artikelnummer erstellen

### Kundenauftrag

```
<?xml version="1.0"?>
<order>
  <salesperson>John Doe</salesperson>
  <item>Production-Class Widget</item>
  <quantity>16</quantity>
  <date>
    <month>1</month>
    <day>13</day>
    <year>2000</year>
  </date>
  <customer>Sally Finkelstein</customer>
</order>
```

Zuliefererauftrag

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    <description>Production-Class Widget</description>
  </item>
  <quantity>16</quantity>
</order>
```

## Artikelnummer erstellen

```
<xsl:template match="/">
  <order>
    <date>...</date>
    <customer>Company A</customer>
    <item>
      <xsl:apply-templates select="order/item"/>
      <quantity>
        <xsl:value-of select="order/quantity"/>
      </quantity>
    </item>
  </order>
</xsl:template>
```

Ergebnis

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    ...
    <quantity>16</quantity>
  </item>
</order>
```

## Artikelnummer erstellen

```
<xsl:template match="item">
  <part-number>
    <xsl:choose>
      <xsl:when test="." = 'Production-Class Widget'>E16-25A</xsl:when>
      <xsl:when test="." = 'Economy-Class Widget'>E16-25B</xsl:when>
      <xsl:otherwise>00</xsl:otherwise>
    </xsl:choose>
  </part-number>
</xsl:template>
```

Ergebnis

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    ...
    <quantity>16</quantity>
  </item>
</order>
```

## Zugriff auf externe Datenbanken



```
<xsl:template match="item">
  <part-number>
    <xsl:choose>
      <xsl:when test="." = "Production-Class Widget">E16-25A</xsl:when>
      <xsl:when test="." = "Economy-Class Widget">E16-25B</xsl:when>
      <xsl:otherwise>00</xsl:otherwise>
    </xsl:choose>
  </part-number>
  ...
</xsl:template>
```

- Kein direkter Zugriff auf eine Datenbank möglich.
- xsl:import erlaubt jedoch importieren externer Templates.
- Lösung: Datenbank → Templates → xsl:import

© Klaus Schild 2003

13

## Artikelbeschreibung übernehmen



### Kundenauftrag

```
<?xml version="1.0"?>
<order>
  <salesperson>John Doe</salesperson>
  <item>Production-Class Widget</item>
  <quantity>16</quantity>
  <date>
    <month>1</month>
    <day>13</day>
    <year>2000</year>
  </date>
  <customer>Sally Finkelstein</customer>
</order>
```

### Zuliefierauftrag

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    <description>Production-Class Widget</description>
    <quantity>16</quantity>
  </item>
</order>
```

© Klaus Schild 2003

14

## Artikelbeschreibung übernehmen



```
<xsl:template match="item">
  <part-number>...</part-number>
  <description>
    <xsl:value-of select="."/>
  </description>
</xsl:template>
```

- Beachte:** "." bezeichnet aktuellen Knoten, hier den Knoten, auf den das Template angewandt wird ("item").

© Klaus Schild 2003

15

## Artikelbeschreibung übernehmen



```
<xsl:template match="item">
  <part-number>...</part-number>
  <description>
    <xsl:value-of select="."/>
  </description>
</xsl:template>
```

### Ergebnis

```
<?xml version="1.0" encoding="UTF-8"?>
<order>
  <date>2000/1/13</date>
  <customer>Company A</customer>
  <item>
    <part-number>E16-25A</part-number>
    <description>Production-Class Widget</description>
    <quantity>16</quantity>
  </item>
</order>
```

© Klaus Schild 2003

16

## Fazit aus dem Beispiel



- Stylesheets nicht einfach zu verstehen und zu debuggen
- Anbindung von Datenbanken umständlich:  
Datenbank → Templates → xsl:import
- Transformation sollte auf dem Server durchgeführt werden, da Kundenauftrag extern nicht sichtbar sein sollte.
- Auf dem Server kann aber für die Transformation eine beliebige andere Programmiersprache benutzt werden.

© Klaus Schild 2003

17