

XML Vortrag

1. Beispiel XML

```
<ADDRESS>
  <FIRSTNAME>Philipp</FIRSTNAME>
  <LASTNAME>Gühring</LASTNAME>
  <STREET>Hebenstreitstr. 16</STREET>
  <ZIP>2602</ZIP>
  <CITY>Neurißhof</CITY>
  <COUNTRY>Austria</COUNTRY>
  <TEL>+43-2628-49103</TEL>
  <FAX/>
  <MOBILE>+43-664-9953109</MOBILE>
  <EMAIL>pg@futureware.at</EMAIL>
  <EMAIL>p.guehring@futureware.at</EMAIL>
  <EMAIL>office@futureware.at</EMAIL>
  <URL>http://www.futureware.at/</URL>
</ADDRESS>
```

2. Begriffe

XML-Dokumente
Wohlgeformtheit
XSLT vs. CSS
XPath
SAX vs. DOM

3. XML Standards

XHTML

- <?xml version="1.0" encoding="iso-8859-1"?>
 - Alle Tags schließen, Tag Struktur bereinigen
 - Kleinschreibung der Tags
 - Anführungszeichen der Tag Attribute
 - & => & und < => <
- tidy -asxml altes.html >neues.xhtml

SVG, MathML, RDF, Docbook, BMEcat, openTRANS, OpenOffice, KOffice, ...

4. Systemanforderungen

Sablotron

<http://www.gingerall.cz/>

Integration in Perl (als Modul), PHP, C, ...

IVI::DB von LivingXML

5. Mythen

- XML hat viel zu viel Overhead
- XML ist zu langsam
- XML ist zu umständlich
- XML ist unsicher

6. XML Anwendungen

Skalierbarkeit: XML > 1 MB

Dateisystem

Jeder Datensatz (optimale Größe: 1 KB) kommt als XML Dokument in eine Datei.

Alle Datensätze einer "Tabelle" kommen in ein Verzeichnis.

Eine Million Dateien in einem Verzeichnis?

adr/1239125:

```
<ADDRESS>
```

```
  <VORNAME>Philipp</VORNAME>
```

```
  <NAME>Gühring</NAME>
```

...

Indizierung: Symbolische Links

ind_adr/Philipp_Gühring_1239125 -> ../sta/1239125

Mehrere Felder? Mehrere gleiche Indizes?

Steuerung: Modell-View-Controller Konzept

Modell: XML

View: XSLT

Controller: Perl + XSLT

7. Workshop

8. Homepage

```
<homepage>
  <!--Hier fangen wir mit dem Menü der Homepage und den Inhalten an -->
  <content>
    <menu>
      <name>XML Vortrag</name>
      <text>
        <h1>XML Vortrag</h1>
        Dies ist ein Vortrag über XML.
        XML ist ein leistungsfähiges Datenverarbeitungs System.
        Im Anschluß daran folgt ein Workshop geben,
        in dem man eine eigene Homepage mit XML machen kann.
      </text>
    </menu>
    <menu id="dv">
      <name>XML Workshop</name>
      <text>
        Starten Sie bitte Ihren Browser, gehen Sie auf www.livingxml.net
und wählen Sie
        rechts unten IVI::DB Admin Demo.
        Dort navigieren Sie bitte in die Linuxwochen Tabelle, und legen
dort ihre eigene Tabelle an.
        Warten Sie dann auf weitere Anweisungen.
      </text>
    </menu>
  </content>

  <!-- Und hier sind die Neuigkeiten -->
  <news>
    <artikel>
      <headline>LivingXML Homepage</headline>
      <link>http://livingxml.net/</link>
    </artikel>
    <artikel>
      <headline>Linuxwochen 2002</headline>
      <link>http://www.linuxwochen.at/</link>
    </artikel>
    <artikel>
      <headline>LivingXML bei den Linuxdays 2001</headline>
      <link>http://www.linuxdays.at/</link>
    </artikel>
    <artikel>
      <headline>IVI::DB Dokumentation</headline>
      <link>http://livingxml.net/ivi/ivi-0.6.pdf</link>
    </artikel>
    <artikel>
      <headline>Neues von den Linuxwochen</headline>
```

```

    <text>Hier bei den Linuxwochen im Tabakmuseum in Wien gibt es
    zahlreiche gute Vorträge</text>
  </artikel>
</news>
</homepage>

```

9. XSLT

XSLT ist die erste XML Programmiersprache.

```

<html xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:param name="menu" />
  <xsl:param name="news" />

  <head>
    <title>XML Workshop</title>
  </head>

  <body>
  <table width="100%">
  <tr>

  <!-- Menü -->
  <td valign="top">
  <table>
  <xsl:for-each select="//homepage/content/*">
  <tr><td><a href="?menu={position()}"><xsl:value-of
  select="name" /></a></td></tr>
  </xsl:for-each>
  </table>
  </td>

  <!-- Seiten Inhalt -->
  <td valign="top">
  <xsl:choose>
    <xsl:when test="$news">
      <xsl:copy-of select="//homepage/news/artikel[number($news)]/text" />
    </xsl:when>
    <xsl:when test="$menu">
      <xsl:copy-of
select="//homepage/content/menu[number($menu)]/text" /><br/><br/>
      <xsl:if test="/homepage/content/menu[number($menu)+1]/name">
        <a href="?menu={$menu+1}">&gt;&gt;
        <xsl:value-of
select="//homepage/content/menu[number($menu)+1]/name" /></a>
      </xsl:if>
    </xsl:when>
    <xsl:otherwise>
      <xsl:copy-of select="//homepage/content/menu[1]/text" />
      <a href="?menu=2">&gt;&gt; <xsl:value-of

```

```

select="//homepage/content/menu[2]/name"/></a>•
  </xsl:otherwise>
</xsl:choose>
</td>

<!-- News Ticker -->
<td valign="top">

<h4>News Ticker</h4>

<xsl:for-each select="//homepage/news/*">
  <a>
    <xsl:choose>
      <xsl:when test="link">
        <xsl:attribute name="href"><xsl:value-of
select="link"/></xsl:attribute>
      </xsl:when>
      <xsl:otherwise>
        <xsl:attribute name="href">?news=<xsl:value-of
select="position()"/></xsl:attribute>
      </xsl:otherwise>
    </xsl:choose>
    <b><xsl:value-of select="headline"/></b>
  </a>
  <br/>
</xsl:for-each>

</td>
</tr>
</table>

</body>
</html>

```

10. Referenzen

<http://www.xml.com/>
 Sablotron (XSLT Parser): <http://www.gingerall.cz/>
 LivingXML: <http://www.livingxml.net/>