

# XML für weltweite Dateiformate

Hubert Partl

## Was ist XML?

XML ist die Abkürzung für "Extensible Markup Language".

XML ist eine Metasprache zur Definition von Markup-Sprachen. So wie HTML mit SGML definiert ist, so kann man mit XML eigene Markup-Sprachen definieren, und künftige Versionen von HTML werden ebenfalls mit XML definiert werden (XHTML).

So wie HTML festgelegt und normiert ist und daher für den weltweiten Austausch und die Übertragung und Verwendung von Web-Pages zwischen vielen verschiedenen Web-Servern und Web-Browsern geeignet ist, so kann man mit XML eigene Datei-Strukturen für verschiedene Zwecke definieren und normieren, die dann ebenfalls von vielen Personen mit vielen verschiedenen Programmen und auf vielen verschiedenen Rechnern verwendet werden können. Mit der Hilfe von Style-Sheets können XML-Dokumente außerdem ebenfalls, so wie HTML-Files, von Web-Browsern dargestellt und ausgedruckt werden.

## Warum XML?

Wofür kann man nun solche mit XML definierte Markup-Sprachen, sogenannte "XML-Applikationen", verwenden? Welchen Zweck kann es haben, mit XML solche Sprachen festzulegen und zu normieren? Welche Vorteile haben solche XML-Anwendungen gegenüber HTML oder anderen Datei-Formaten?

- Mit XML kann man die logische Bedeutung von Daten, Informationen und Texten definieren-ähnlich wie die Tabellen- und Spalten-Bezeichnungen in Datenbanken und Tabellenkalkulationen.
- XML ermöglicht im Gegensatz zu HTML die Definition eigener oder zusätzlicher "Befehle" (Tags)-ähnlich wie bei der Definition von Macros in der Textverarbeitung.
- XML-Applikationen eignen sich als Plattform- und Software-unabhängiges Austausch-Format für Daten zwischen verschiedenen Programmen und Rechnern - ähnlich wie RTF für Texte, CVS für Tabellen, EDI für kommerzielle Anwendungen -

aber in einem einheitlichen, allgemein verwendbaren, Hersteller-unabhängigen Format.

Außerdem ist die Syntax von XML so streng festgelegt, dass XML-Anwendungen wesentlich einfacher, bequemer und effizienter von Programmen weiter verarbeitet werden können als HTML-Files.

## Markup und Darstellung

Die HTML-Befehle beschreiben eigentlich nur, in welcher Art und Weise die Textteile strukturiert werden sollen (Überschrift, Absatz, Liste, Tabelle, Normal-schrift, Fettschrift usw.).

### Beispiel

Der folgende HTML-Code

```
<p>
Hubert Hans <b>Partl</b>
<br>
Muthgasse 18
<br>
A-1190 Wien
<br>
geb. 8. März 1949
</p>
```

bewirkt eine Darstellung wie

**Hubert Hans Partl**  
Muthgasse 18  
A-1190 Wien  
geb. 8. März 1949

Mit XML kann man Tags definieren, die die Bedeutung der Informationen angeben. Mit einer geeigneten DTD (Document Type Definition) kann die obige Information also in einem XML-Dokument in der folgenden Form enthalten sein:

```
<person id="25264" >
  <vorname>Hubert</vorname>
  <vorname>Hans</vorname>
  <zuname>Partl</zuname>
  <titel>Dr.</titel>
  <adresse>Muthgasse 18</adresse>
  <plz>A-1190</plz>
  <ort>Wien</ort>
  <geburtstag>
    <tag>8</tag>
    <monat>März</monat>
    <jahr>1949</jahr>
  </geburtstag>
</person>
```

Mit einem geeigneten Style-Sheet bewirkt das XML-Dokument genau die selbe Darstellung wie das obige HTML-Dokument:

**Hubert Hans Partl**  
Muthgasse 18  
A-1190 Wien  
geb. 8. März 1949

Mit einem anderen Style-Sheet kann dasselbe XML-Dokument aber auch so dargestellt werden (als Visitenkarte):

### Hubert Hans Partl

Muthgasse 18  
A-1190 Wien

oder so (in einer Liste):

Partl, Dr. Hubert Hans (1949)

## Verarbeitung in Programmen

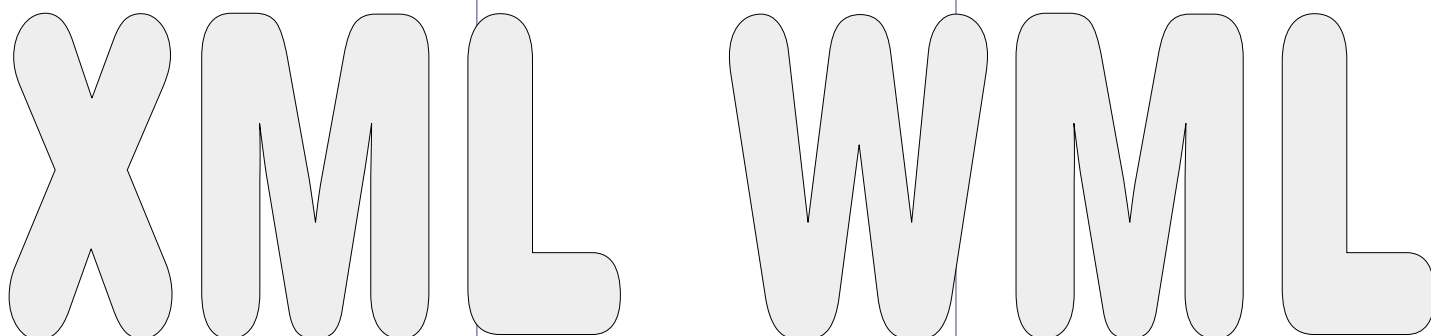
XML-Dokumente eignen sich nicht nur für die Darstellung in Web-Browsern und das Ausdrucken auf Papier, sondern auch für die weitere Verarbeitung in Programmen, in denen die logische Bedeutung der Informationen eine Rolle spielt.

### Beispiel

Eine Suche nach dem Wort "März" innerhalb von Absätzen <p> in HTML-Files der obigen Struktur würde nicht nur die Personen liefern, die im März geboren sind, sondern auch den Rektor mit dem Namen **Leopold März** sowie alle Personen, die in der **Märzstraße** wohnen.

Eine Suche nach dem Wort "März" innerhalb der in den XML-Files mit <geburtstag> bezeichneten Elemente würde hingegen wirklich nur die Personen liefern, die in diesem Monat Geburtstag haben.

Außerdem kann man die in den XML-Files beschriebenen Personen leicht nach den einzelnen Datenfeldern sortieren, also z.B. nach der Postleitzahl oder nach dem Geburtsjahr, oder das Durchschnittsalter berechnen oder andere Verarbeitungen der Daten vornehmen.



# WAP und WML für Web-Pages auf Handys

Hubert Partl

## Was sind WAP und WML?

WML (Wireless Markup Language) ist das Format, in dem Hypertext-Informationen für Geräte mit sehr kleinen Displays definiert werden können, also für Mobil-Telefone ("Handys"), PDAs, Palm-top-Computer, Auto-Navigationsgeräte und dergleichen. WML erfüllt eine ähnliche Funktion wie HTML, das sich besser für Geräte mit größeren Displays eignet, also für Notebooks, Desktop-PCs, Workstations, Fernsehapparate und dergleichen und für die Druckausgabe auf Papier.

WML baut nicht direkt auf HTML auf sondern auf XML (Extensible Markup Language). WML ist ähnlich wie HTML, aber nicht mit HTML kompatibel.

WAP (Wireless Application Protocol) ist das besonders effiziente Protokoll, mit dem WML-Seiten vom Handy abgefragt und an das Handy übertragen werden. Im allgemeinen wird WAP zwischen dem Handy und einem WAP-Gateway verwendet, und die Übertragung zwischen dem Web-Server und dem WAP-Gateway erfolgt über das Internet mit dem Protokoll HTTP.

## Warum WML?

Wenn Sie Ihre Web-Informationen nicht nur mit HTML sondern auch mit WAP und WML verfügbar machen, erreichen Sie nicht nur die klassischen Internet-Benutzer, die einen PC oder Fernsehapparat zum "Internet-Surfen" verwenden, sondern auch die wesentlich stärker wachsende Anzahl von Personen, die ein Mobil-Telefon ("Handy") oder ähnliches

Gerät besitzen und darauf Informationen abfragen können.

HTML hat zwar den Vorteil, dass sich die in den Web-Pages enthaltenen Informationen durch das Prinzip des logischen Markup automatisch an die Fenstergröße und Schriftgröße des Client-Rechners anpassen, egal ob es sich um einen Notebook oder PC oder eine Workstation mit großem oder kleinem, niedrig oder hoch auflösendem Bildschirm handelt

Allerdings funktioniert diese automatische Anpassung nur innerhalb eines Bereiches von wenigstens einigermaßen vergleichbaren Fenster- und Papiergrößen. Typische PC- und Fernseh-Bildschirme bieten Platz für etwa 200 Wörter in ca. 20 Zeilen, Papierblätter für etwa 600 Wörter in ca. 60 Zeilen.

Für Geräte mit sehr kleinen Displays, die nur etwa 10 bis 20 Wörter in 2 bis 5 Zeilen darstellen können, muß man die Informationen im allgemeinen extra aufbereiten, in einem stark gekürzten und wesentlich kompakteren Format. Da solche Geräte meistens auch nur über geringere Rechen-Ressourcen verfügen als PCs oder Workstations, müssen außerdem ein Format und ein Protokoll verwendet werden, die möglichst effizient verarbeitet werden können. Deshalb werden für diese Zwecke nicht HTML und HTTP sondern WML (das auf XML aufbaut) und WAP verwendet. XML und damit auch WML hat strengere Regeln als HTML und erlaubt dadurch eine effizientere Verarbeitung.

Wenn man eine Information für *alle* Endbenutzer mit *allen* möglichen Geräten zur Verfügung stellen will, dann muss man

sie auf dem Web-Server in zwei Versionen verfügbar halten:

- die **komplette** Version für große Bildschirme und Printouts als HTML-File oder als XML- oder XHTML-File
- und die **gekürzte** Version für kleine Displays als WML-File.

In den meisten Fällen wird man weder das HTML-File noch das WML-File händisch erstellen, sondern beide automatisch und dynamisch aus den in einer Datenbank oder einem Workflow-System gespeicherten Informationen generieren.

Es gibt auch spezielle WAP-Gateways, die versuchen, bestehende HTML-Informationen, so gut das geht, automatisch in WML umzuwandeln. Meistens genügt eine solche automatische Umwandlung aber nicht, weil die Informationen für die kleinen Displays anders (kürzer) formuliert und anders strukturiert werden müssen.

Zum Testen der WML-Files gibt es WML-Simulatoren, die wie ein Web-Browser auf PCs unter MS-Windows laufen, sodass man die WML-Files gleich am PC testen kann, ohne ein Handy zu benutzen, ohne teure WAP- und Telefonkosten.

## Ausführlichere Informationen

- WAP-Forum  
<http://www.wapforum.org/>
- Phone-Page  
<http://www.phone.com/>
- Nokia-Page  
<http://www.nokia.com/corporate/wap/>
- WML-Einführung  
<http://www.boku.ac.at/html/inf/>



## XML-Unterstützung in Web-Browsern

Künftige Web-Browser werden XML-Files direkt am Bildschirm darstellen und am Drucker ausdrucken können, wenn mit einem Style-Sheet definiert ist, wie die einzelnen XML-Elemente dargestellt werden sollen. Erste Ansätze dafür gibt es im MS Internet Explorer Version 5 (mit einer Vor-Version von XSL Style-Sheets oder mit CSS) sowie in Netscape Version 6 und Mozilla (mit CSS Style-Sheets).

Zu diesem Zweck werden am Web-Server sowohl das XML-File mit dem Inhalt der Information als auch das XSL- oder CSS-File mit den Layout-Angaben abgespeichert - so ähnlich wie im Textsatzsystem LaTeX, wo der Inhalt im TEX-File

und das Layout im STY-File definiert werden, und so wie dort hat man auch hier die Möglichkeit, den selben Inhalt wahlweise in verschiedenen Layouts darzustellen, z.B. für große und kleine Bildschirme und für Schwarz-weiß- und Farb-Drucker.

Wenn man die Informationen für alle Benutzer verfügbar machen will, also auch für diejenigen, die noch ältere Web-Browser verwenden, muss man sie am Web-Server (zusätzlich) im normalen HTML-Format zur Verfügung stellen - am besten mit einem Umwandlungsprogramm, das die XML-Files mit der Hilfe von Style-Files automatisch in Standard-HTML umwandelt. Und wenn die Informationen auch mit Handy-Telefonen und dergleichen erreichbar sein sol-

len, dann gleich auch noch im WML-Format (Wireless Markup Language).

## Ausführlichere Informationen

- XML-Spezifikationen  
<http://www.w3.org/XML/>
- Newsgruppe  
comp.text.xml
- XML FAQ  
<http://www.ucc.ie/xml/>
- XML in der Praxis  
<http://www.mintert.com/xml/>
- XML Kurz-Info  
<http://www.boku.ac.at/html/inf/>