Seminar
„Informatik für Biologen“
WS 04/05

15.01.2005

XML
eXtensible Markup Language

Andreas Hiergeist

Folgende Ausarbeitung beschäftigt sich mit der Entstehung, der Struktur, der Funktion, einiger Einsatzmöglichkeiten und Anwendungen (insbesondere im wissenschaftlichen Bereich) von XML. Die Ausarbeitung soll dabei nur einen kurzen Einblick über die Möglichkeiten, die XML bietet geben.

Was ist XML ?

XML steht für die Abkürzung „Extensible Markup Language“, was im Deutschen soviel wie „Erweiterbare Auszeichnungs-Sprache“ bedeutet. XML ist insofern erweiterbar, weil es die Möglichkeit bietet – im Gegensatz zu HTML beispielsweise – eigene „Auszeichnungen“ und damit auch neue Auszeichnungssprachen zu definieren. XML wird deshalb auch als Metasprache (d.h. eine Sprache zur Beschreibung von Sprachen) bezeichnet. Zusammengefasst bedeutet dies, XML liefert die Regeln, die bei der Definition von Dokumenttypen angewendet werden müssen.

Unter einer Auszeichnugnssprache, („Markup-Language“) versteht man eine Sprache, die der strukturierten Darstellung von Informationen beziehungsweise Dokumenten dient. Ein Beispiel für eine solche Auszeichnungssprache wäre HTML (Hypertext Markup Language), die speziell zur Darstellung von Dokumenten im „World Wide Web“ konzipiert wurde. Die Auszeichnung erfolgt mit Hilfe von Tags:

<StartTag>ausgezeichneter Text</EndTag>

Die Entstehung von XML

Grundsätzlich ist die Idee für die Entwicklung einer Metasprache zur Strukturierung von Dokumenten nicht neu. Bereits im Jahr 1986 wurde ein internationaler Standard zur Beschreibung der Struktur und des Inhalts von verschiedenen Arten von elektronischen Dokumenten entwickelt (IBM). Dies wurde erforderlich um einen einfachen Datenaustausch zwischen Unternehmen zu ermöglichen. SGML bietet diese

Plattformunabhängigkeit, ist jedoch in ihrer Anwendung sehr komplex und stellt für verschiedene Probleme zahlreiche Optionen zur Verfügung, sodass sie vor allem für Organisationen geeignet ist, die anspruchsvolle Standards für ihre Dokumente benötigen. Und weniger für einen einfachen Datenaustausch (beispielsweise übers Internet).

Speziell für das Internet wurde 1989 HTML (Hypertext Markup Language) mit Hilfe von SGML definiert. HTML ist vor allem für die Darstellung von Seiten im Internet ausgelegt und daher in seinen Möglichkeiten zum Datenaustausch begrenzt.

Durch die Weiterentwicklung des Internet und die steigende Anforderungen an Internetseiten wurde für bestimmte Zwecke die Entwicklung einer neuen Sprache nötig. So entstand 1998 XML durch die Arbeit einer Arbeitsgruppe des „Word Wide Web Consortiums“, einer Standardisierungs-Organisation für das Web. XML ist dabei als Teilmenge von SGML zu sehen, wobei die nützlichsten Teile in XML übernommen, wobei die komplexen, optionalen Teile wegelassen wurden. Dies bietet zusätzlich den Vorteil, dass einige für SGML aufwändig programmierte Anwendungen ebenfalls mit XML verwendbar sind.

Ausgehend von XML haben sich mittlerweile zahlreiche weitere Markup-Sprachen (XML-Anwendungen) wie WML für WAP-Handys oder MathML zur Beschreibung mathematischer Ausdrücke entwickelt.

Folgende Abbildung zeigt die Beziehung von SGML, HTML und XML zueinander.

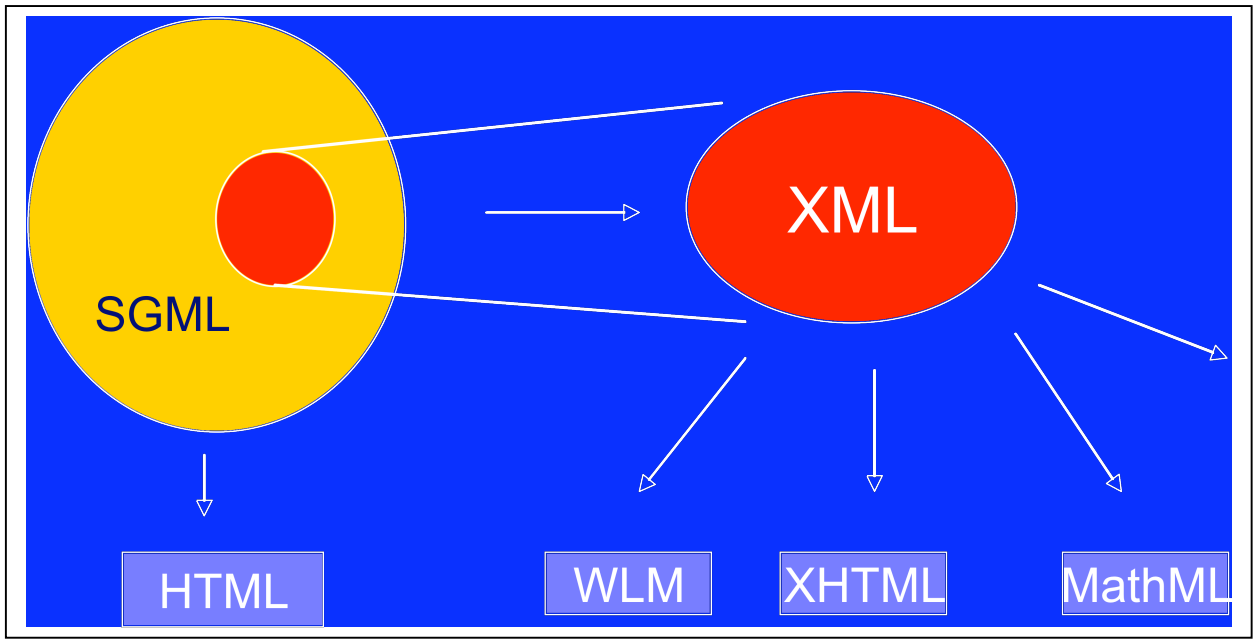

Abbildung 1: Entwicklung von XML

XML und HTML

Um die wesentlichen Merkmale (Vorteile) von XML besser zu verstehen, soll zunächst der Vergleich mit HTML herangezogen werden.

```
<h2>
Max Mustermann
</h2>
<p>
Industriestr. 49<br>
<b>89231 Neu-Ulm</b>
</p>
```

HTML

```
<Adresse>
  <Vorname>Max</Vorname>

<Nachname>Mustermann<Nachname>
  <Straße>Industriestr.</Straße>
  <Hausnummer>49</Hausnummer>
  <PLZ>89231</PLZ>
  <Ort>Neu-Ulm</Ort>
</Adresse>
```

XML

Wie man an obigem Beispiel erkennen kann, dient HTML in erster Linie der Darstellung von Informationen, wohingegen XML (durch die Verwendung „semantischer Tags“) eine Beschreibung von Informationen liefert. Dies bietet die Möglichkeit, dass XML-Inhalte maschinell zugänglich gemacht, ausgewertet und verarbeitet werden können.
Ein weiterer Vorteil von XML ist die Erweiterbarkeit, indem man eigene Auszeichnungselemente (Tags) definieren kann. Durch die Vielzahl der dadurch entstehenden Möglichkeiten ist eine Anpassung an spezifische Bedürfnisse möglich, was vor allem im Bereich der Dokumentverarbeitung von Bedeutung ist.

Die Bestandteile von XML

Eine gurndlegende Idee von XML ist die klare Trennung von Inhalt, Struktur und Darstellung. Diese Trennung manifestiert sich im Wesentlichen in drei unterschiedlichen Dateitypen.

- **XML-Dateien**, mit dem eigentlichen Inhalt (Daten) des Dokuments
- **DTD-Dateien** (bzw XML-Schema) für die Struktur (Grammatik)
- **XSL-Dateien** (bzw. XSLT-Dateien) für die Darstellung (Form) des Dokuments

Das XML Dokument

Den Aufbau eines XML-Dokuments beschreibt folgende Abbildung. Ein XML-Dokument, das den eigentlichen Inhalt des Dokuments enhält lässt sich in Prolog und Rumpf gliedern. Im Prolog findet man Verarbeitungsanweisungen (PI = Processing Instructions) zur systemspezifische Angaben ein ein Anwendungsprogramm zu übermitteln.

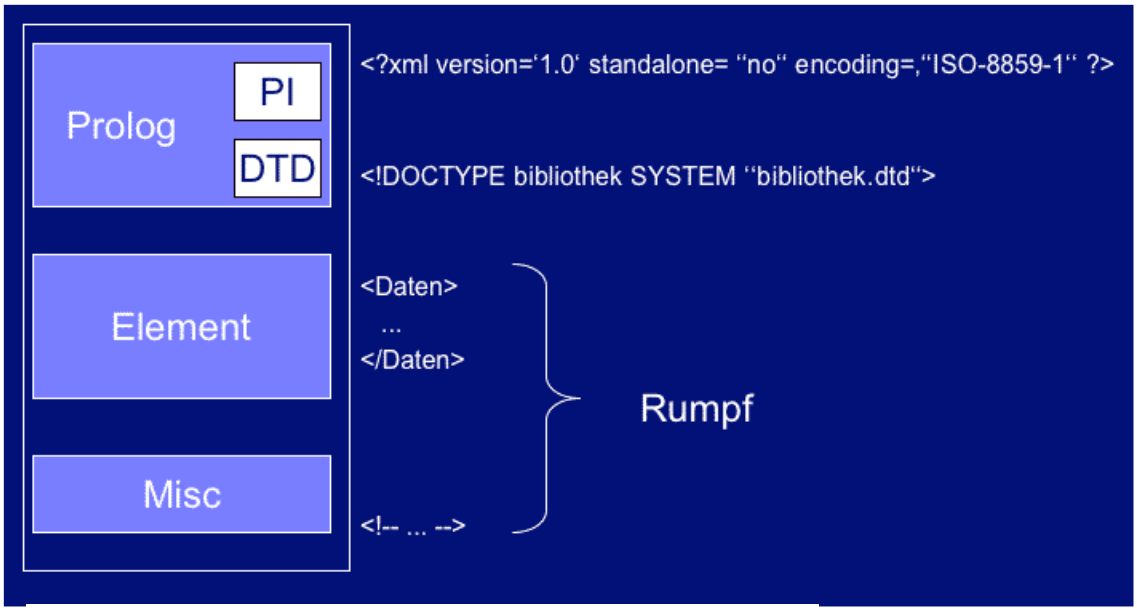

Abbildung 2: Aufbau eines XML-Dokuments

Der Rumpf enthält die eigentlichen Daten des Dokuments, die in Tags eingebettet sind. Außerdem dürfen in einem XML-Dokument an beliebiger Stelle Kommentare (<!--...-->) stehen.

Die Regeln zur Erstellung eines XML-Dokuments sollen an einer exemplarischen XML-Datei erläutert werden. Diese ist in folgender Abbildung dargestellt. Als Beispiel wurde eine Bibliothek vereinfacht dargestellt. Dieses Beispiel lässt sich jedoch beliebig erweitern.

```
<?xml version=‘‘1.0‘‘?>
 <bibliothek>
        <buch erscheinungsjahr=‘‘1997‘‘>
             <titel>Molekulare Genetik</titel>
             <autor>
                  <vorname>Rolf</vorname>
                  <nachname>Knippers</nachname>
             </autor>
             <ISBN>3134770075</ISBN>
             <verlag>Thieme</verlag>
        </buch>
        <buch erscheinungsjahr=‘‘1999‘‘>
            <titel>Organische Chemie</titel>
            <autor>
                 <vorname>Harold</vorname>
                 <nachname>Hart</nachname>
             </autor>
             <ISBN>3527303790</ISBN>
             <verlag>WileyVCH</verlag>
        </buch>
   </bibliothek>
```

Wie man am Beispiel des XML-Dokuments erkennen kann sind sämtliche Datensätze, welche sich aus den einzelnen Elementen (z.B. <buch>...</buch>) zusammensetzen **einem** Wurzelelement (<bibliothek>...</bibliothek>) untergeordnet.

Neben den Elementen kommen Attribute (<buch erscheinungsjahr...>) mit Attributwerten (z.B. ’’1997’’) vor. Attribute ermöglichen eine schnellere Suche nach bestimmten Inhalten.

Allgemein ist durch die XML-Regeln eine strenge Syntax vorgegeben. So müssen geöffnete Tags stets wieder geschlossen oder leere Elemente besonders gekennzeichnet werden (<leeresElement/>). Desweiteren wird zwischen Groß- und Kleinschreibung unterschieden. Liegt eine korrekte Verschachtelung vor (Elementinhalt enhält neue korrekte Tags oder Text/Daten) so spricht man von einem „wohlgeformten“ XML-Dokument und die logische Struktur beschreibt eine Baumstruktur. Diese strikte XML-Syntax macht XML-Dokumente leichter durch Programme verarbeitbar.

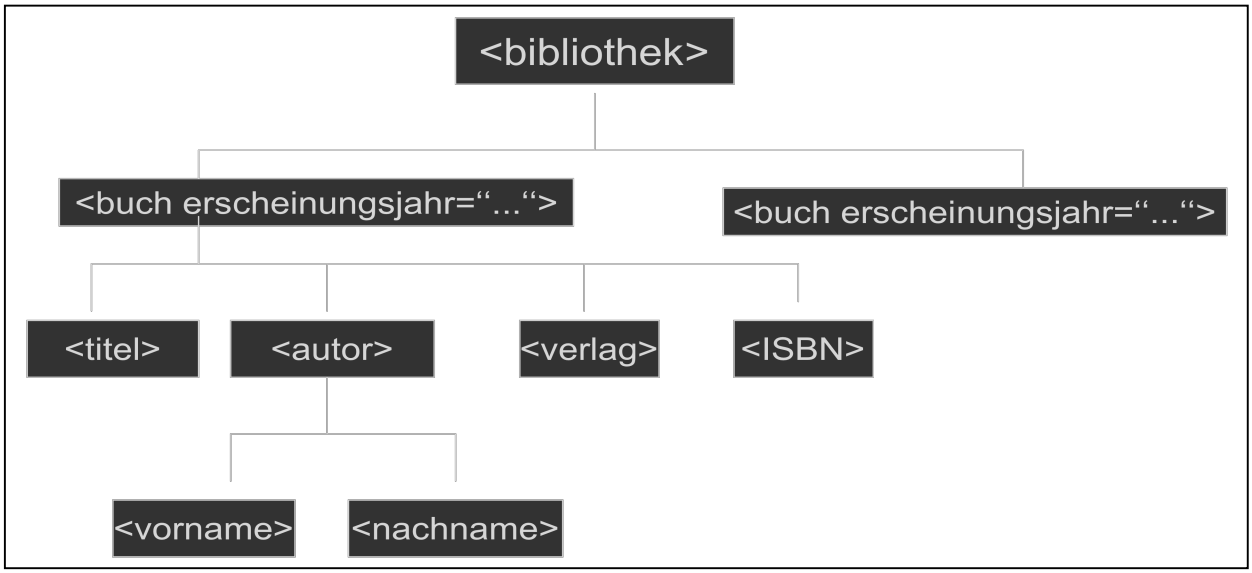

Abbildung 3: Darstellung der Baumstruktur eines XML-Dokuments

Die Dokumenttyp-Definition (DTD)

In einer DTD werden die Regeln für den Aufbau von Dokumenten geliefert. Sie beschreibt also die XML-Grammatik. Für das Beispiel obiger Bibliothek ist hier nun die entspredchende DTD dargestellt.

```
<!ELEMENT bibliothek (buch+)>
  <!ELEMENT buch (titel, autor+, verlag, ISBN)>
  <!ATTLIST buch erscheinungsjahr CDATA>
      <!ELEMENT titel (#PCDATA)>
      <!ELEMENT verlag (#PCDATA)>
      <!ELEMENT autor (vorname, nachname)>
          <!ELEMENT vorname (#PCDATA)>
        <!ELEMENT nachname (#PCDATA)>
      <!ELEMENT ISBN (#PCDATA)>
```

In der DTD ist also die Verwendung der Elemente (<!ELEMENT ...>) und Attribute (<!ATTLIST ...>) im XML-Dokument beziehungsweise deren Verschachtelung und

Vorkommen vorgeschrieben. Dadurch wird die Syntax, Struktur und Bedeutung der verwendeten Tags definiert.
Durch Verwendung von DTDs kann also eine Klasse von gleichartigen XML-Dokumenten bei der Erstellung einer Markup-Sprache abgedeckt werden, wodurch DTDs nicht für jede XML-Datei neu erstellt werden müssen.
DTD-Dateien können sowohl außerhalb des XML-Dokuments (extern) stehen und dort im Prolog eingebunden werden (<!DOCTYPE .....>) oder innerhalb der XML-Datei (intern) stehen.
Eine Alternative zu DTDs stellt das sogenannte XML-Schem dar, das im Gegensatz zur DTD in XML geschrieben ist. Besonders bei vielschichtigen Anforderungen bietet XML-Schema einige Vorteile gegenüber DTDs, sind jedoch wesentlich komplexer aufgebaut.

XSL(T) zur Darstellung des Dokuments

Grundsätzlich trägt der Inhalt des XML-Dokuments die eignetliche Information und kann bereits durch Anwendungen weiterverarbeitet werden. Ist jedoch eine formatierte Darstellung des Dokuments nötig/erwünscht, stehen sogenannte StyleSheets zur Verfügung. Hauptsächlich kommt in XML (neben den CSS (Cascading StyleSheets) die XSL (extensible StyleSheet Language) zum Einsatz. Man unterscheidet dabei zwischen der XSL/FO (Formatting Objects), wodurch eine Formatierung bzw. Erstellung eines Seitenlayouts möglich ist und XSLT (=XSL-Transformation), das die Möglichkeit der Transformation in andere Formate (wie z.B. PDF, HTML ...) bietet.
Mit Hilfe von XSLT ist es möglich, ein XML-Dokument für verschiedene Zielmedien aufzubereiten. So können aus einer Datenquelle beispielsweise durch Verwendung verschiedener XSLT-Stylesheets sowohl Web-Inhalte in Form von HTML als auch für mobile Endgeräte (WML) oder multimediale CD-ROMs aufbereitet werden.
Dieses als „Multichannel Publishing“ bezeichnete Einsatzgebiet von XML ist in folgender Abbildung dargestellt.

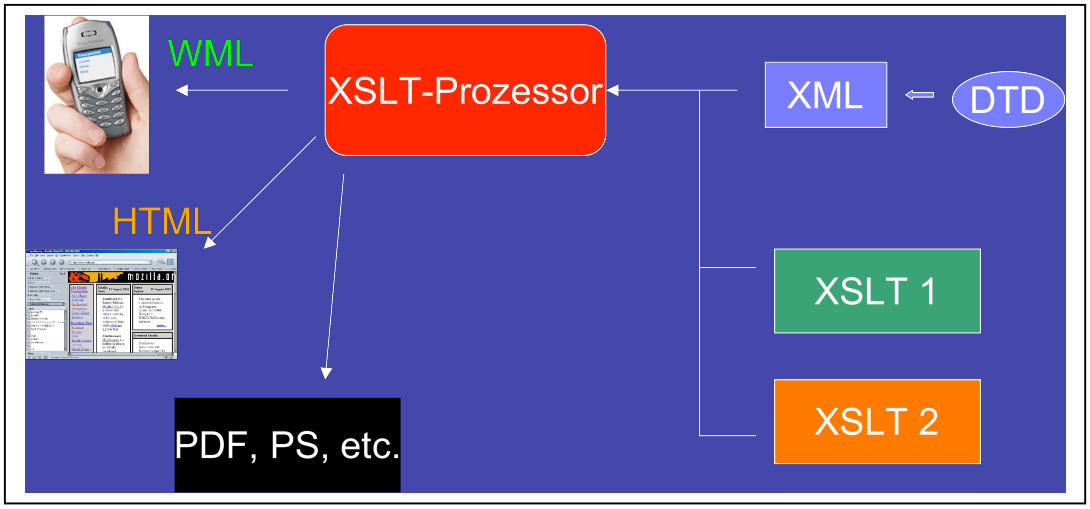

Abbildung 4: Möglichkeit des Multichannel-Publishing durch XSLT

Einsatzmöglichkeiten von XML

Wie bereits beschrieben lassen sich Daten durch Verwendung von StyleSheets oder Transformation (mit XSLT) auf unterschiedlichen Medien (Web, Print, CD-Rom etc..) präsentieren.

Durch die Einfachheit und Erweiterbarkeit von XML, wodurch sich XML auf die gewünschte Anwendung anpassen lässt, ergeben sich außerdem vielfältige Einsatzmöglichkeiten.

XML wird vor allem für die Weitergabe von Daten innerhalb von Firmen (z.B. für den Zwischenbetrieblichen Austausch von Geschäftsdokumenten) oder zwischen Forschungseinrichtungen (Workflow) benutzt.

Desweiteren bietet XML als Metasprache die Grundlage zur Definition neuer, speziell auf die jeweilige Anwendung angepasster Sprachen. Diese werden hautpsächlich dazu genutzt, Daten für bestimmte Zwecke auszutauschen oder zu präsentieren.

Beispiele hierfür wären Sprachen wie VRML (Virtual Reality Markup Language) oder SMIL (Synchronised Multimedia Integration Langugage) im Multimedia-Bereich, WML für die Darstellung von Dokumenten auf mobilen Endgeräten oder Sprachen im Bereich der Wissenschaften.

XML als Format zum Austausch von Daten:

Gerade im Bereich der Wissenschaft ist es ist ein Austausch von Daten (zum Beispiel innerhalb einer Arbeitsgruppe), das Veröffentlichen von Daten, Datenanalyse (z.B. mit spezieller Software) oder das Sammeln von Daten (z.B. aus dem Internet) von großer Bedeutung. Das Problem, das sich dabei ergibt ist, dass beispielsweise verschiedene Software (z.B von konkurrierenden Unternehmen) zur Datenanalsyse unterschiedliche proprietäre Dateiformate verwenden.
Auch hier bietet XML als Plattform- und Software-unabhängiges Austauschformat für Daten eine Lösung. Die strenge festgelegte Syntax von XML-Dokumenten ermöglicht außerdem eine einfache, bequeme und effiziente Weiterverarbeitung durch Programme.

XML-Anwendungen

Unter dem Begriff „XML-Anwendungen“ versteht man spezifische Markup-Sprachen, die mit Hilfe von XML definiert wurden, also die XML-Syntax benutzen.
Durch die Möglichkeit der Definition eines eigenen Vokabulars in XML ist es möglich, diese XML-basierte Sprachen auf eigene Bedürfnisse anzupassen. Einige Beispiele aus dem Bereich der (Bio)Wissenschaften sollen hier vorgestellt werden.

- **MathML** (Mathematical Markup Language)

  Zur Beschreibung und Darstellung mathematischer Ausdrücke

- **CML** (Chemical Markup Language)

  Verwaltung verschiedener Informationen (atomare, molekulare, kristallografische..) von Molekülen. Darstellung der moleküle mit eigenem Browser

- **CellML** (Cellular Markup Language)

  Entwicklung und Austausch biologischer Modelle und Beschreibung mathematischer Modelle zur Darstellung von Vorgängen der Elektrophysiologie, Signaltransduktion und Stoffwechselwegen

- **BSML** (Bioinformatic Sequence Markup Language)

  Darstellung („Genome Viewer“) und Bearbeitung von Sequenz-informationen von DNA, RNA und Proteinen aus Datenbanken (z.B. NCBI)

- **KGML** (KEGG Markup Language)

  Darstellung von Stoffwechselvorgängen (inkl. Metabolite, Enzyme...)

- **SBML** (Systems Biology Markup Language)

  Repräsentation biochemischer Modelle: Zellkompartimente, biochemische Reaktionen (ähnlich CellML)

- **Taxonomic Markup Language**

  Speicherung und Austausch von taxonomischen Beziehungen zwischen Organismen

Literatur:

- http://www.w3.org/XML/
- http://www.xml-web.de
- http://www.xml-slt.de
- http://www.xml-cml.org
- http://g4xml.com
- http://www.bmsl.org
- http://www.cellml.org
- http://www.sbw-sbml.org
- http://www.info.uni-karlsruhe.de/~i44www/lehre/XML-WS-01-02/
- http://industry.ebi.ac.uk/~alan/XMLWorkshop/
- http://xml.coverpages.org/bioml.html

- http://www.ib.hu-berlin.de/~wumsta/sgml/index.html
- http://www.w3schools.com/xml/default.asp
- http://www.informatik.uni-jena.de/%7Esack/WS0405/webtechnologien-materialien.htm
- Achard, F, XML, bioinformatics and data integration, Bioinformatics Vol 17 no.2, 2001 pp. 115-125
- Helmut Erlenkötter, XML, Rohwolt Taschenbuch, 2003