

# XML

## Eine praktische Einführung

# Übersicht

- Einführung
- HTML - Codebeispiel
- XML
  - Dreiteilung in Struktur, Daten, Darstellung
  - Beispiel
- EDI
- Abgrenzung XML - HTML
  - Übersicht SGML/XML/HTML
  - Links
- Ausblick

# HTML - Beispiel

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Home</TITLE>
```

```
</HEAD>
```

```
<BODY BACKGROUND="SatelliteBackground.gif">
```

```
<B>
```

```
<FONT SIZE="+3"> Welcome <BR>
```

```
</FONT>
```

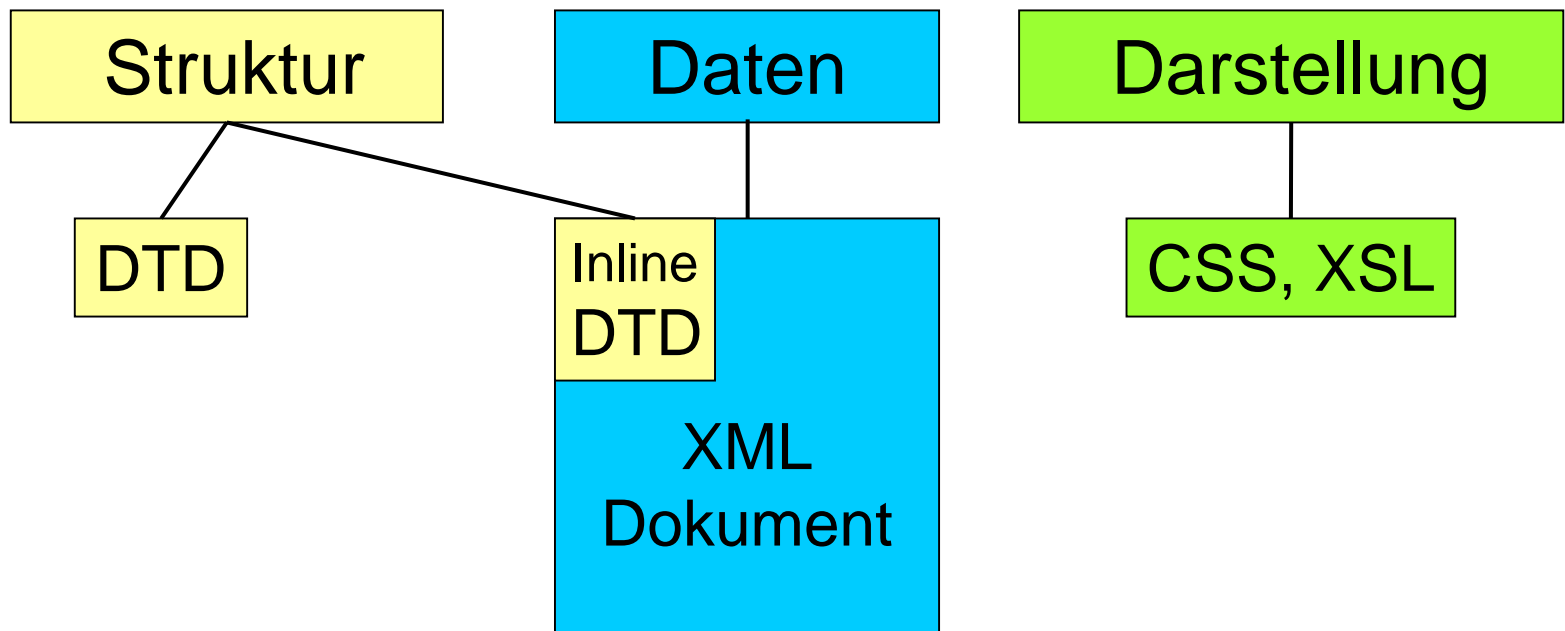
```
</B>
```

```
<A HREF="what_s_new.html"> In addition</A>
```

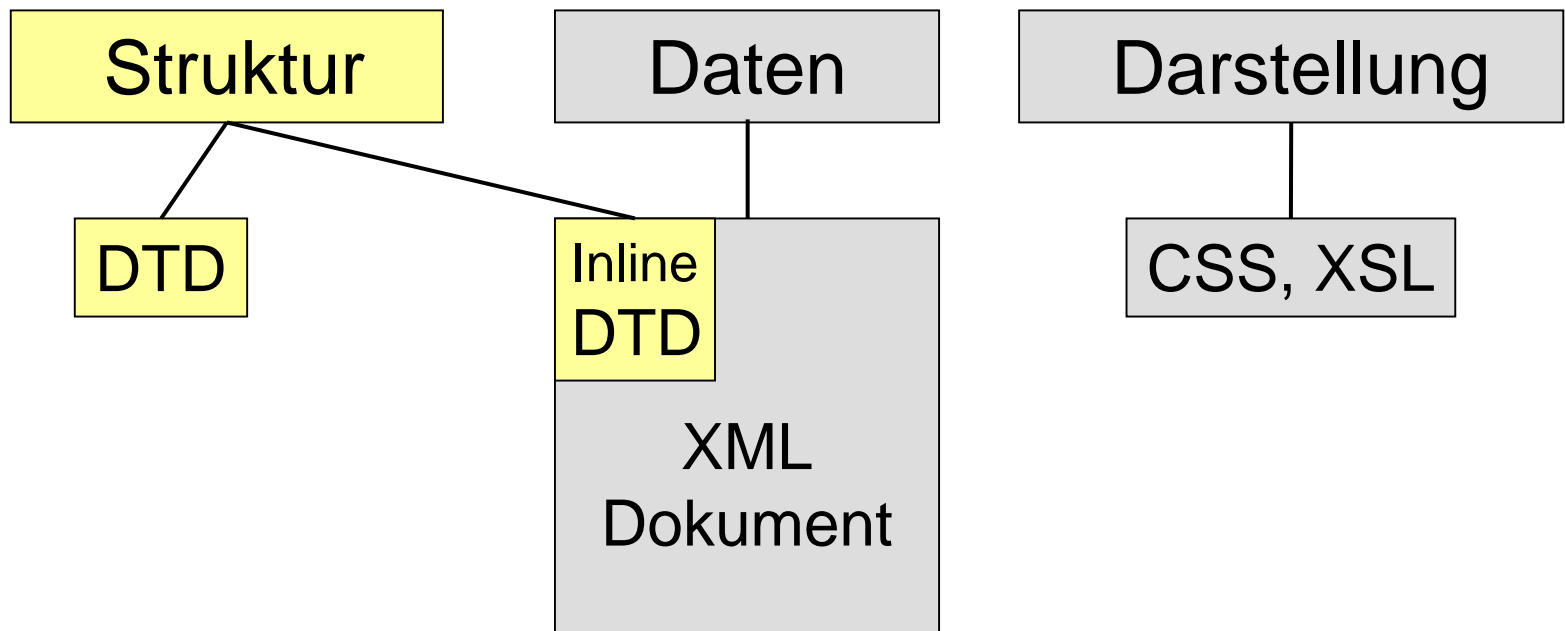
```
</BODY>
```

```
</HTML>
```

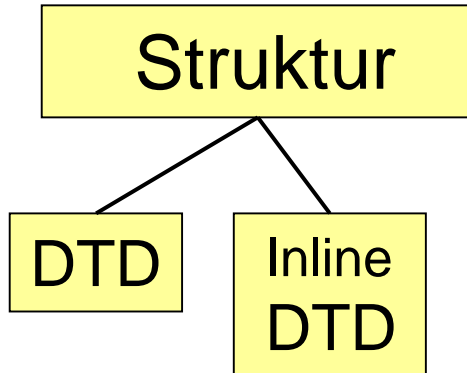
# Dreiteilung



# Dreiteilung



# DTD: Übersicht



In der **Document Type Definition** werden **Elemente** des XML-Dokumentes definiert.

Die **Inline DTD** überschreibt eventuelle Definitionen aus der externen DTD.

Beispiel:

```
<!ELEMENT KATALOG (PRODUKT+)>
```

```
<!ELEMENT PRODUKT (NAME+, INFO+, PREIS+, NOTIZ?)>
```

# DTD: Elemente

- XML-Dokumente sind in **verschachtelten Elementen** strukturiert, diese werden in der DTD definiert.
- Anhand der DTD kann die **Wohlgeformtheit** (syntaktische Korrektheit) des XML-Dokumentes **geprüft** werden.
- Der **Zeichenvorrat** der Elemente kann **eingeschränkt** werden.
- Elemente können Attribute haben.

# DTD: Elemente

Wie **häufig ein Element** vorkommen darf kann festgelegt werden:

in XML:

Einmal oder Keинmal	?
Einmal oder Mehrmals	+
Keинmal oder Mehrmals	*

Beispiel:

```
<!ELEMENT PRODUKT (NAME+, INFO+, PREIS+, NOTIZ?)>
```



# DTD: Attribute

Jedem Element können Attribute zugeordnet werden, es gibt 3 Attributtypen:

- **String-Attribut**, beliebig vielen Zeichen
- **Token-Attribut**, ein oder mehrere XML-Token
- **Aufzählungs-Attribut**, Liste möglicher Werte

Beispiel:

```
<!ELEMENT INFO (#PCDATA)>
```

```
<!ATTLIST INFO
```

```
    GROESSE (S | M | L | XL) "M" #REQUIRED
```

```
    FARBE    CDATA #REQUIRED>
```

# Inline-DTD: Listing

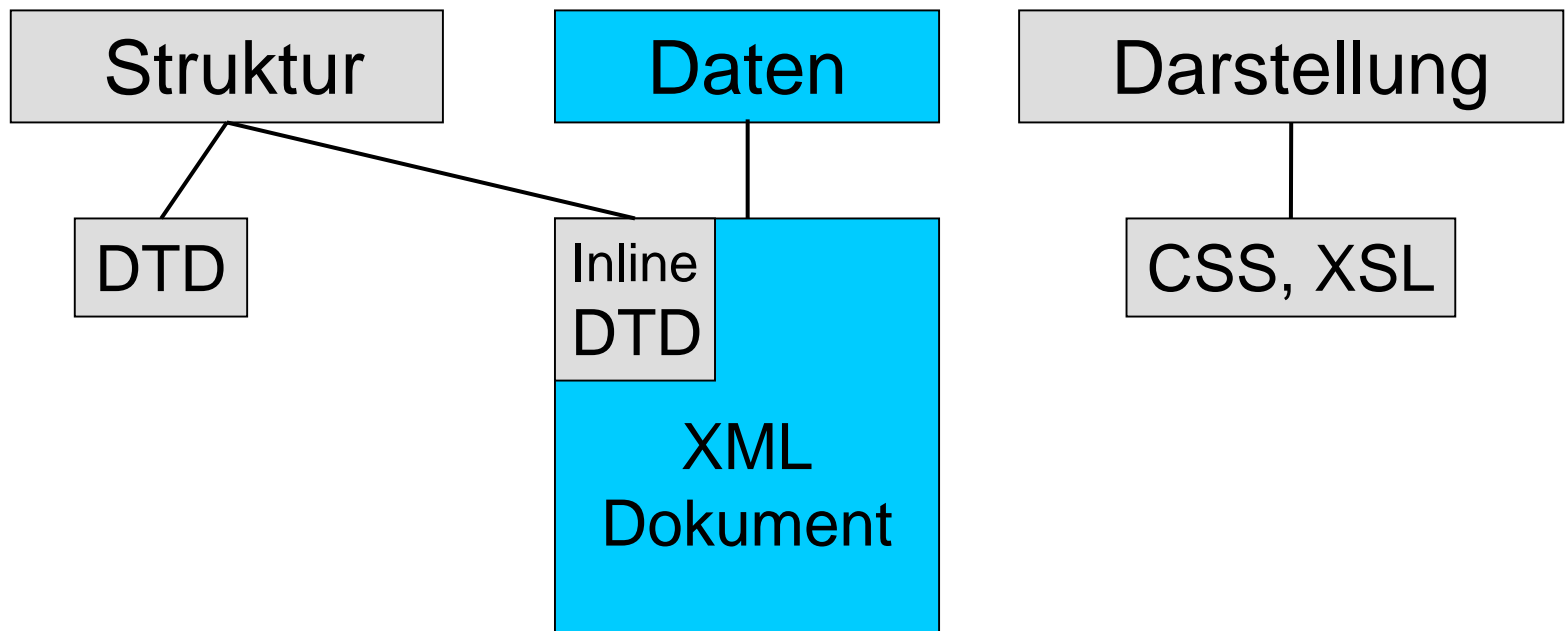
```
<!DOCTYPE KATALOG [  
  <!ELEMENT KATALOG (PRODUKT+)>  
  <!ELEMENT PRODUKT (NAME+, INFO+, PREIS+, NOTIZ?)>  
  <!ATTLIST PRODUKT NAME CDATA #REQUIRED>
```

```
<!ELEMENT INFO (#PCDATA)>  
<!ATTLIST INFO  
          GROESSE (S | M | L | XL) "M" #REQUIRED  
          FARBE    CDATA          #REQUIRED>
```

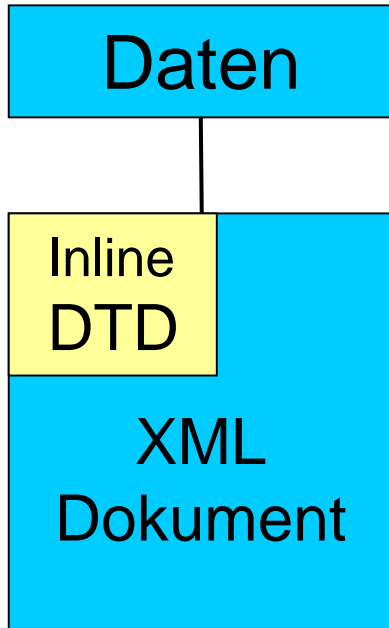
```
<!ELEMENT PREIS (#PCDATA)>  
<!ATTLIST PREIS GROSSHANDEL NMTOKEN #REQUIRED  
                EINZELHANDEL NMTOKEN #REQUIRED  
                STEUERN      NMTOKEN #IMPLIED>
```

```
<!ELEMENT NOTIZ (#PCDATA)>  
>
```

# Dreiteilung: Daten



# XML-Dokument: Übersicht



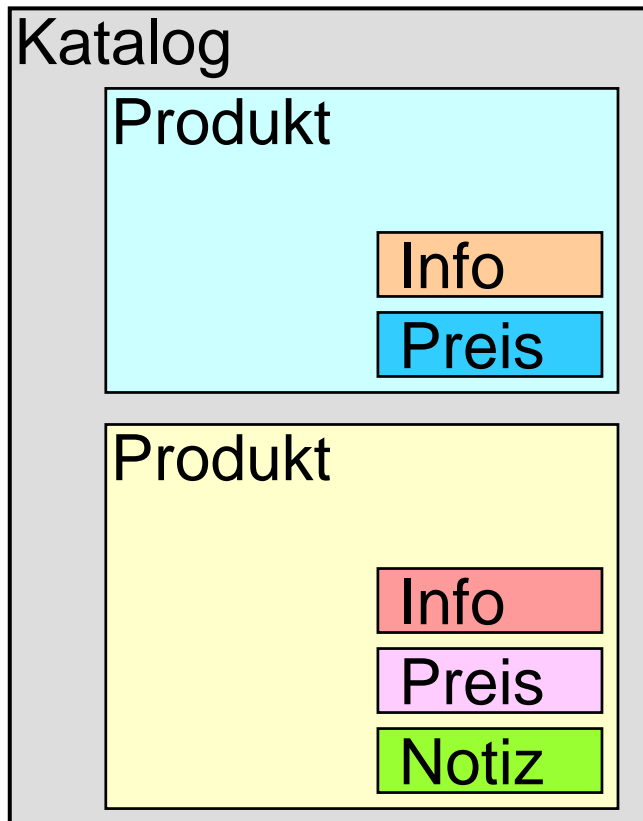
Das XML-Dokument enthält in **strukturierter Form** Daten und Informationen.

Der Inline DTD und der externe DTD enthalten die **Strukturbeschreibung**.

Das restliche XML Dokument enthält die **Daten** in der durch die **Strukturbeschreibung festgelegten Struktur**

# XML-Dokument: Aufbau

Informationen werden in geschachtelten Elementen abgelegt:



```
<KATALOG>
```

```
  <PRODUKT NAME="T-shirt">
```

```
    <INFO GROESSE="XL"
```

```
      FARBE    ="ROT" />
```

```
    <PREIS EINZELHANDEL="29.95"
```

```
      GROSSHANDEL="9.95"
```

```
      STEUER="2.56" />
```

```
  <NOTIZ>
```

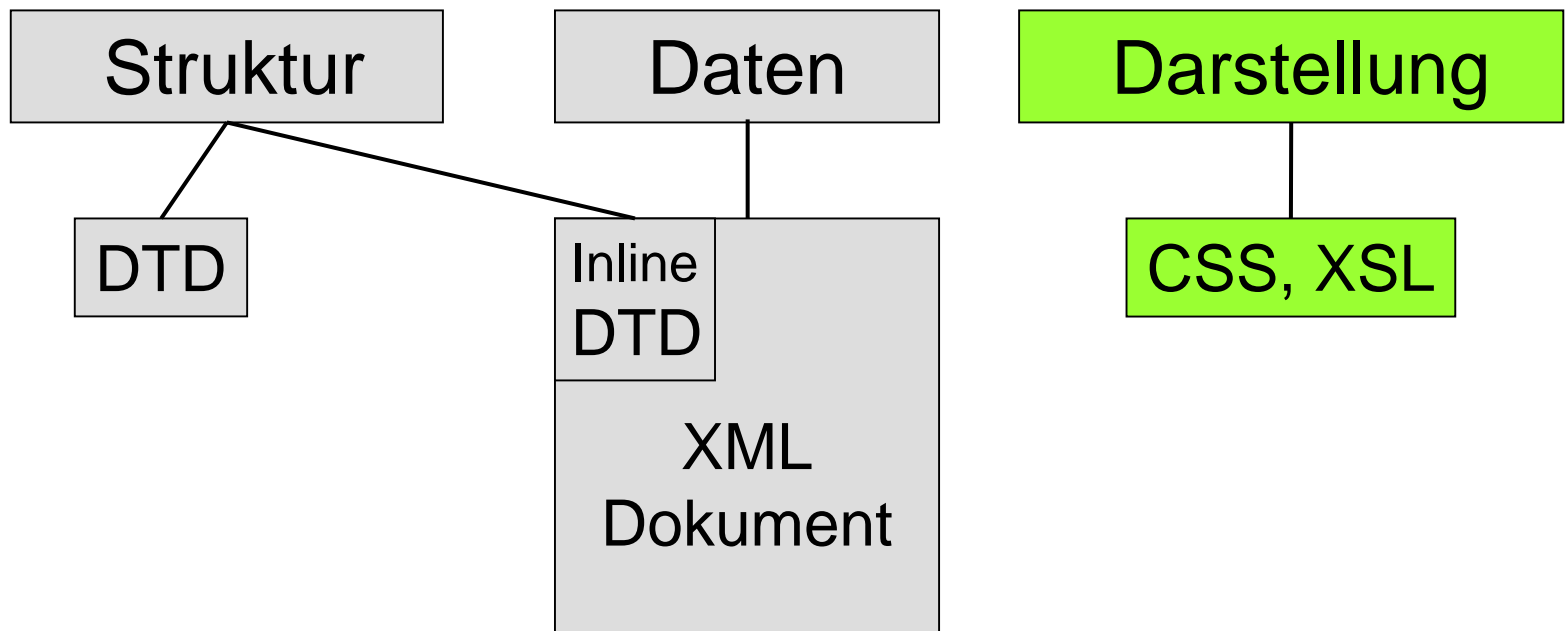
```
    NEU im Sortiment!
```

```
  </NOTIZ>
```

```
  </PRODUKT>
```

```
</KATALOG>
```

# Dreiteilung: Darstellung



# Darstellung: Übersicht

Darstellung

CSS, XSL

Die **Darstellung** der hinterlegten Strukturen und Daten, erfolgt durch **Formatierungen und Skripte**. Die Formatierungen und Skripte, liegen in **externen Dateien** vor (z.B. CSS-Datei).

Beispiel:

```
KATALOG { display: block;  
          font-family:arial,sans-serif;font-size: 20pt}
```

```
PRODUKT { display: block; border-top: 2pt solid black; }
```

# Beispiel

Im Microsoft Internet Explorer erzwingt die Verarbeitungs-Anweisung:

```
<?xml:stylesheet href="Katalog.css" type="text/css"?>
```

die Darstellung unter Zuhilfenahme des **Cascaded Style Sheet**.

- » [Internet Explorer](#)
- » [CSS-Datei Listing](#)
- » [XML-Datei Listing](#)



# EDI

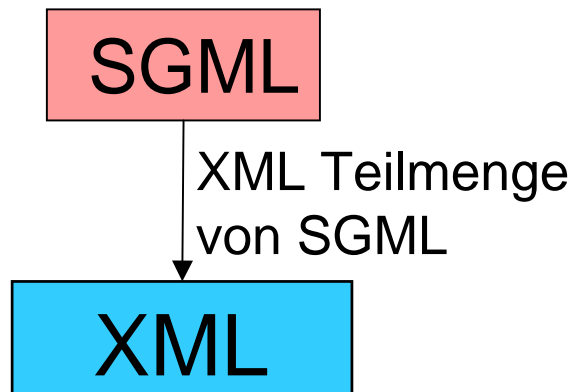
Wenn eine DTD festgelegt wurde, können Daten sowohl:

- **validiert**, als auch
- **ausgetauscht** werden

Microsoft entwickelt zur Zeit einen Übersetzer (BizTalk-Server), der EDIFACT-Nachrichten nach XML konvertiert.<sup>1</sup>

1) nach: [edi-change] Seite 38 f.

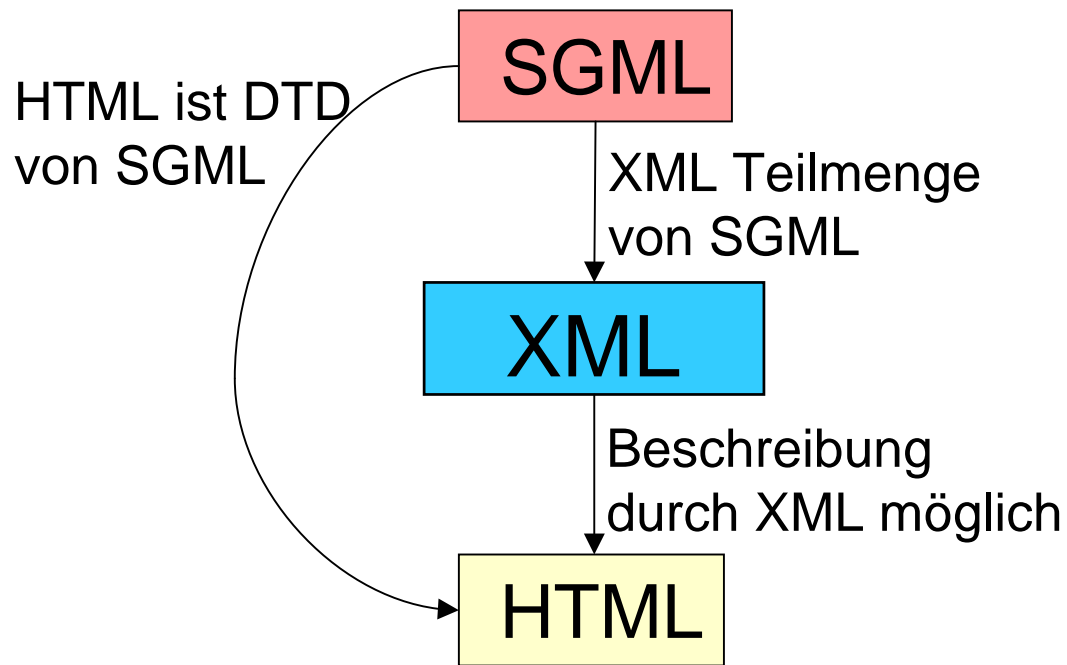
# Abgrenzung: XML



SGML ist eine Sprache zur Beschreibung von Markup-Sprachen.

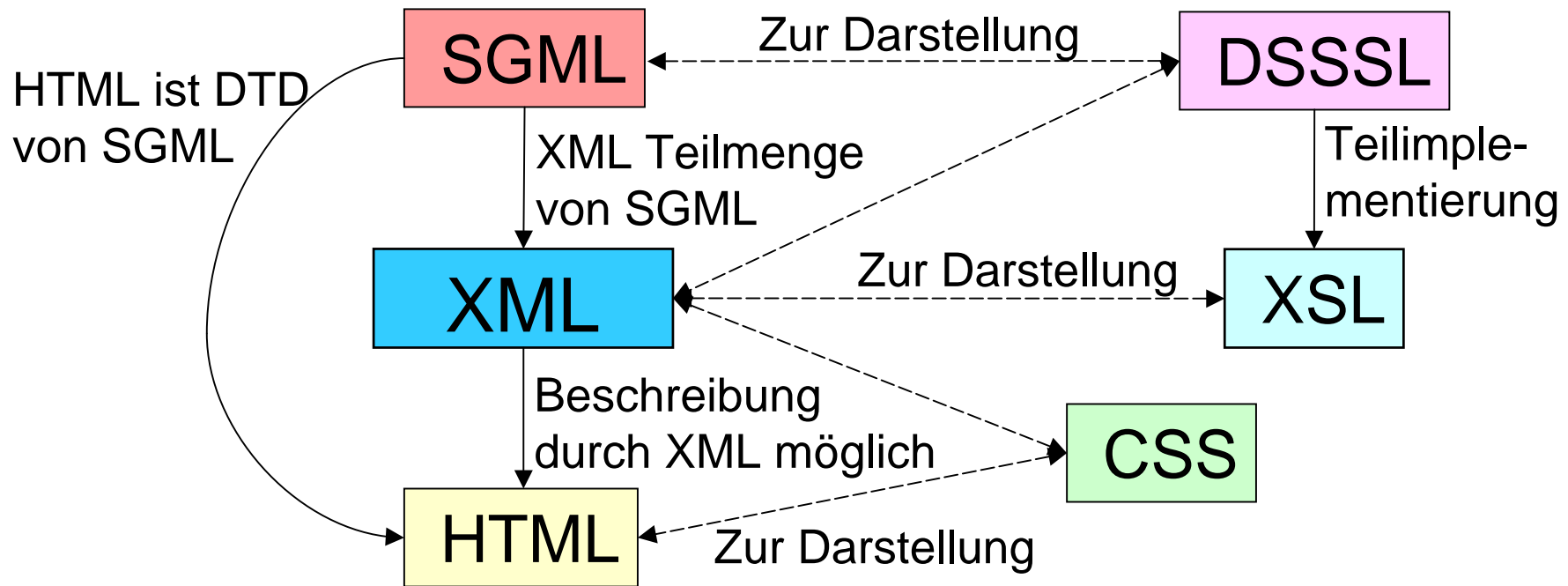
XML kann in SGML beschrieben werden

# Abgrenzung: XML



HTML ist eine spezielle, beschriebene Struktur.  
Es ist nicht möglich, neue Strukturen zu definieren

# Abgrenzung: XML



Formatierungen für verschiedene Ausgabemedien durch die Dokumenten - Darstellungs - Sprachen

In Anlehnung an: [Mintert] Abbildung 1

# Links: URI

Ein **Unified Resource Identifier** (URI) existiert in 2 verschiedenen Konzepten:

- Identifikation anhand des Ortes (**URL**)
- Identifikation anhand des Namens (**URN**)

Die Syntax des URI besteht aus einem „**Namen** und einem **namensspezifischen Teil**“<sup>1</sup>  
<Name>:<namensspezifischer Teil>

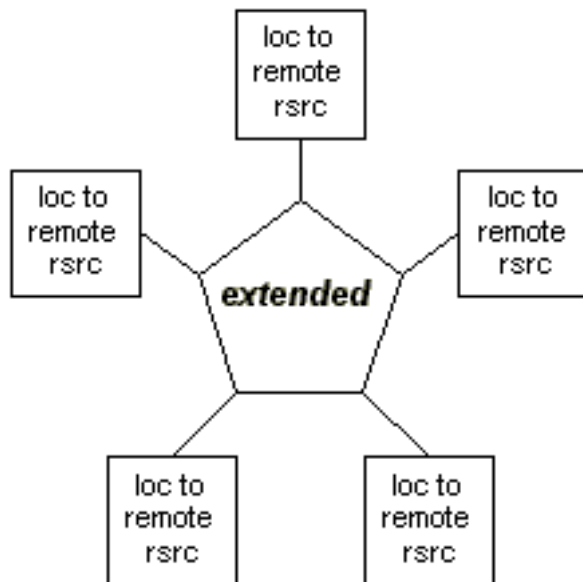
Die durch den Namen spezifizierte Anwendung interpretiert den namensspezifischen Teil.

1) nach: [RFC 2396] Seite 10ff.

# Links: XLink

XLink ist ein Vorschlag zur Realisierung von Links in XML.

Es gibt **simple Links**, die ähnlich den unidirektionalen Links in HTML sind.



Mächtiger sind **extended Links**.

Grafik aus: [<http://www.w3.org/TR/xlink/>]

# Link: extended

Extended Links haben u. a. folgende Attribute:

- **type** = Typdeklaration, hier „extended“
- **href** = Verweis, URI
- **title** = Beschreibender Titel
- **role** = Elementbeschreibung des gelinkten Objekts
- **arc** = Anweisung zum traversieren

# Link: role

**Role** spezifiziert den Typ, des gelinkten Objektes

Beispiel in der DTD:

```
<!ELEMENT course EMPTY>
```

```
<!ATTLIST course
```

```
    xlink:type      (extended)    #FIXED "extended"
```

```
    xlink:href      CDATA          #REQUIRED
```

```
    xlink:role     NMTOKEN      #FIXED "course"
```

```
    xlink:title     CDATA          #IMPLIED>
```

Beispiel im XML-Dokument:

```
<course
```

```
    xlink:href="/courses/cs101.xml"
```

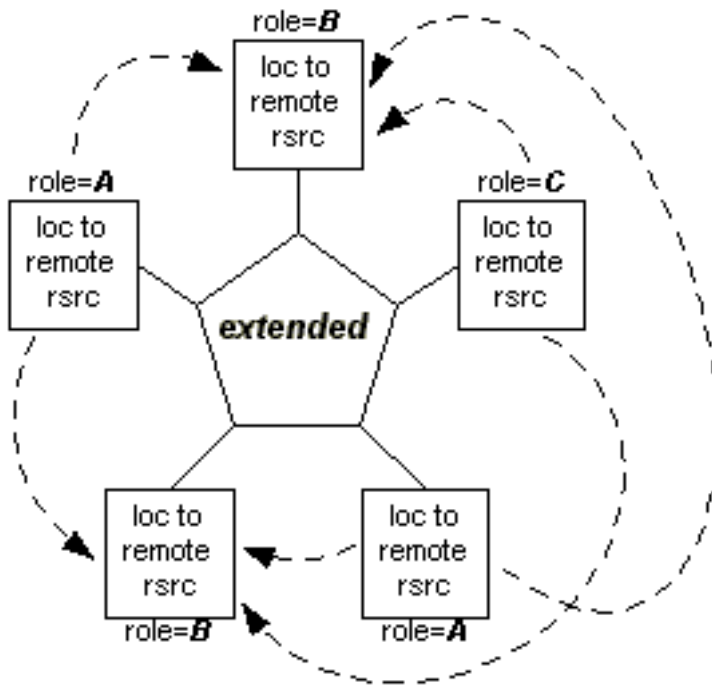
```
    xlink:title="CompSci 101" />
```

Beispiel aus: [<http://www.w3.org/TR/xlink/>]



## Link: arc

**Arc** ermöglicht die Angabe von Verknüpfungen unter bestimmten Voraussetzungen:



**Beispiel:**

```

<!-- go = arc-type -->
<go xlink:from="A" xlink:to="B" />
<go xlink:from="C" xlink:to="B" />
    
```

Grafik und Beispiel aus: [<http://www.w3.org/TR/xlink/>]

# Link: Ausblick

- XLink ist zur Zeit nur ein **Vorschlag**
- Nur für **wenige, spezielle** Anwendungen bisher implementiert
- Sehr **mächtiges Konzept** durch „extended Links“

# Aus-/ Überblick

Zutaten



Kochrezept &  
Kochwerkzeuge



Kuchen



XML-  
Dokument

+  
anderes  
XSL / CSS

=  
verändertes  
ausgegebenes  
Dokument

# Ausblick

- Viele Anwendungsmöglichkeiten durch die enthaltene **Struktur-Information**
- Zukunftsorientiert, durch **Trennung** der Darstellungs- /Formatierungsebene
- Zum **Datenaustausch** schon gebräuchliches Format (implementiert von z.B. Oracle, Informix)
- Beliebig **erweiterbare Strukturen**
- Leider **wenig standardisierte DTD's**

**ENDE**



# Listing: XML-Demo

```
<!-- Die Verarbeitungsanweisung weist IE an, eine CSS-Datei zur Darstellung zu verwenden -->
<?xml:stylesheet href="Katalog.css" type="text/css"?>
<KATALOG>
  <PRODUKT>
    <NAME>T-Shirt</NAME>
    <INFO>  <GROESSE>XL</GROESSE>
            <FARBE>Gruen</FARBE>

    </INFO>
    <PREIS> <EINZELHANDEL>29,95</EINZELHANDEL>
            <GROSSHANDEL>9,95</GROSSHANDEL>
            <STEUER>2,20</STEUER>

    </PREIS>
    <NOTIZ>NEU im Sortiment</NOTIZ>
  </PRODUKT>
  gekürzt...

</KATALOG>
```



# Listing: CSS-Datei

```
KATALOG { display: block; font-family:arial,sans-serif;font-size: 20pt}
```

```
PRODUKT { display: block; border-top: 2pt solid black; }
```

```
NAME    { display: block; margin-left:24pt; text-align:center;  
          text-decoration:underline font-weight: bold;}
```

```
INFO    { display: block; margin-left:24pt; padding-top: 6pt;}
```

```
PREIS   { display: block; margin-left:60pt; list-style-type: lower-roman;  
          padding-top: 6pt;}
```

```
    EINZELHANDEL { display: block;text-indent: 10pt;}
```

```
    GROSSHANDEL { display: block;display: none;}
```

```
    STEUER      { display: block;text-indent: 10pt;}
```

```
NOTIZ   { display: block; margin-left:24pt; color: red; padding-top: 6pt;}
```

