

# DAIMLERCHRYSLER

## **Der XML-Schemaansatz als Voraussetzung datenorientierter XML- Vokabulare der zweiten Generation**

Mario Jeckle

DaimlerChrysler Forschungszentrum Ulm

[mario.jeckle@daimlerchrysler.com](mailto:mario.jeckle@daimlerchrysler.com)

[mario@jeckle.de](mailto:mario@jeckle.de)

[www.jeckle.de](http://www.jeckle.de)

# Gliederung

## I Einführung und Motivation

- Historie der Extensible Markup Language
- Strukturelle Grundkonzepte der Extensible Markup Language
- Datenorientierte Auszeichnungssprachen

## II XML Schema

- Notwendigkeit und Ansätze
- Entwicklungslinien und Standards
- Die XML Sprachfamilie und XML Vokabulare der zweiten Generation

## III Anwendungen, Kritik am Ansatz und zukünftige Entwicklung

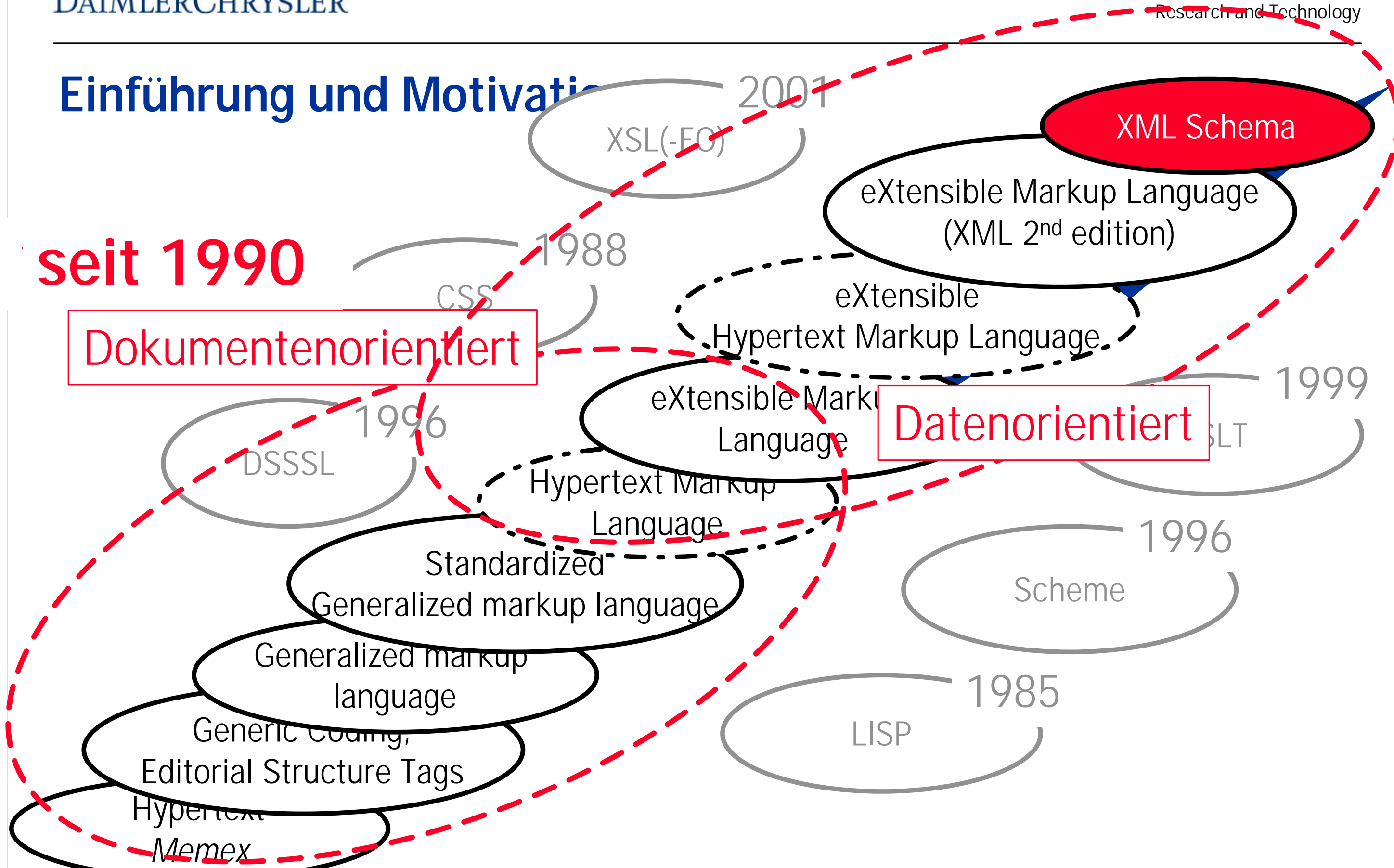
- XML Vokabulare
- Vokabularerzeugung
- Ausblick: Die nächste XML Schema Generation

# Einführung und Motivation

seit 1990

Dokumentenorientiert

Datenorientiert



## Einführung und Motivation

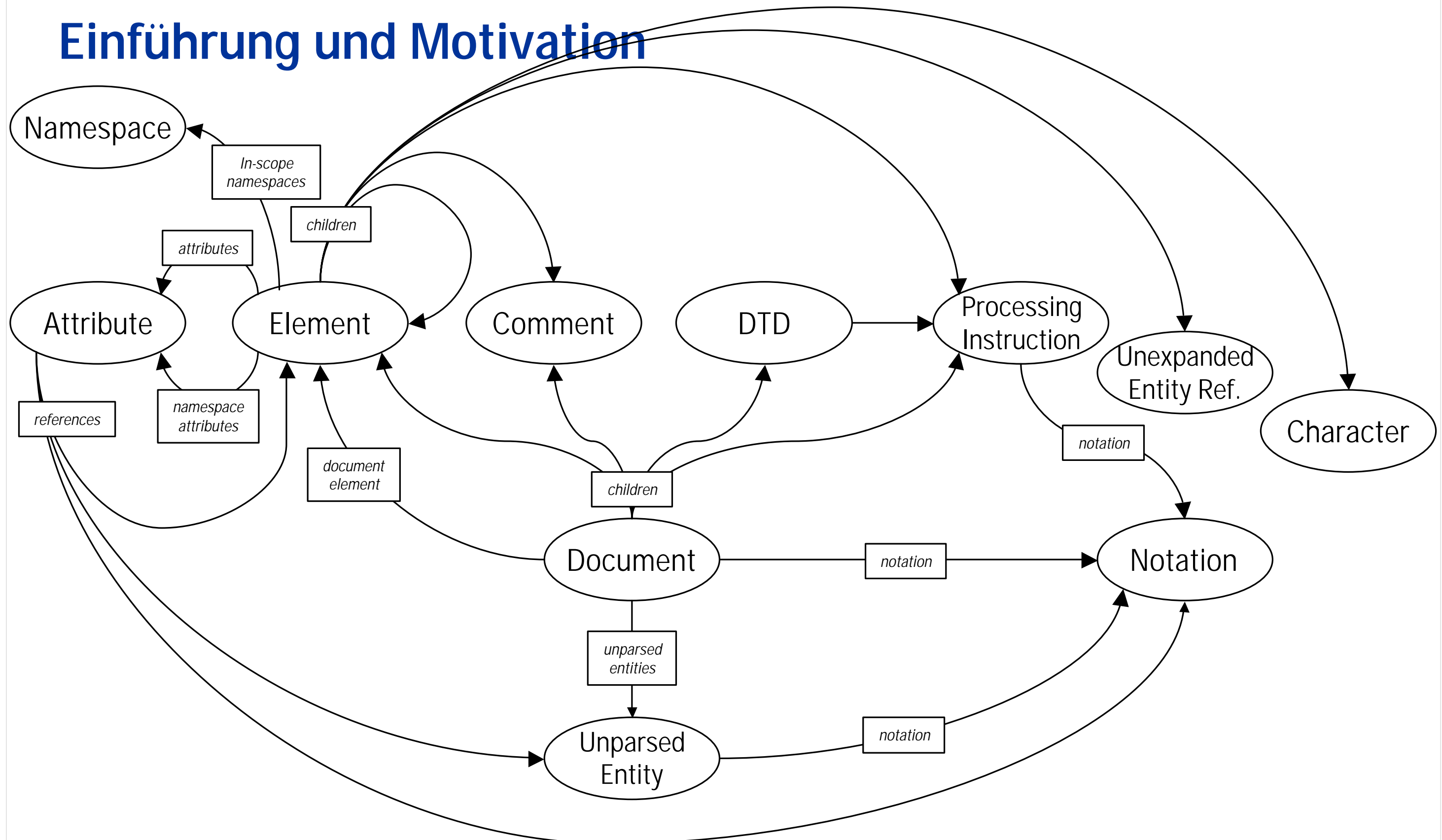
```
<?xml version="1.0" encoding="UTF-8"?>
<Vortrag>
  <Titel>Der XML-Schemaansatz als Voraussetzung
    datenorientierter XML-Vokabulare der zweiten Generation</Titel>
  <Veranstaltung datum="2002-06-18">
    <Name>Informatik Kolloquium Uni Stuttgart</Name>
  </Veranstaltung>
  <Referent>
    <Name>Mario Jeckle</Name>
    <Firma>DaimlerChrysler Research and Technology</Firma>
    <URL>http://www.jeckle.de</URL>
    <E-Mail>mario.jeckle@daimlerchrysler.com</E-Mail>
  </Referent>
</Vortrag>
```

## Einführung und Motivation

### Der XML Information Set:

- Definiert strukturellen Aufbau einer XML-Quelle
  - ... als *abstrakte Datenmenge*
  - ... als Definitionssammlung
- Jedes wohlgeformte XML-Dokument besitzt einen InfoSet
  - ... auch wenn es nicht als Datei vorliegt
  - ... daher ist diese Spezifikation Basis weiterer W3C-Spezifikationen (XML-Schema, XMLP/SOAP, WSDL, ...)
- W3C Recommendation seit 2001-10-24
  - ... definiert keinerlei Programmierparadigmen-spezifische Schnittstellen
  - ... operiert auf abstrakten Baumstrukturen
- Wird durch XML-Schemavalidierungsprozeß
  - ... auf strukturelle Konformität geprüft
  - ... mit Typinformation dekoriert

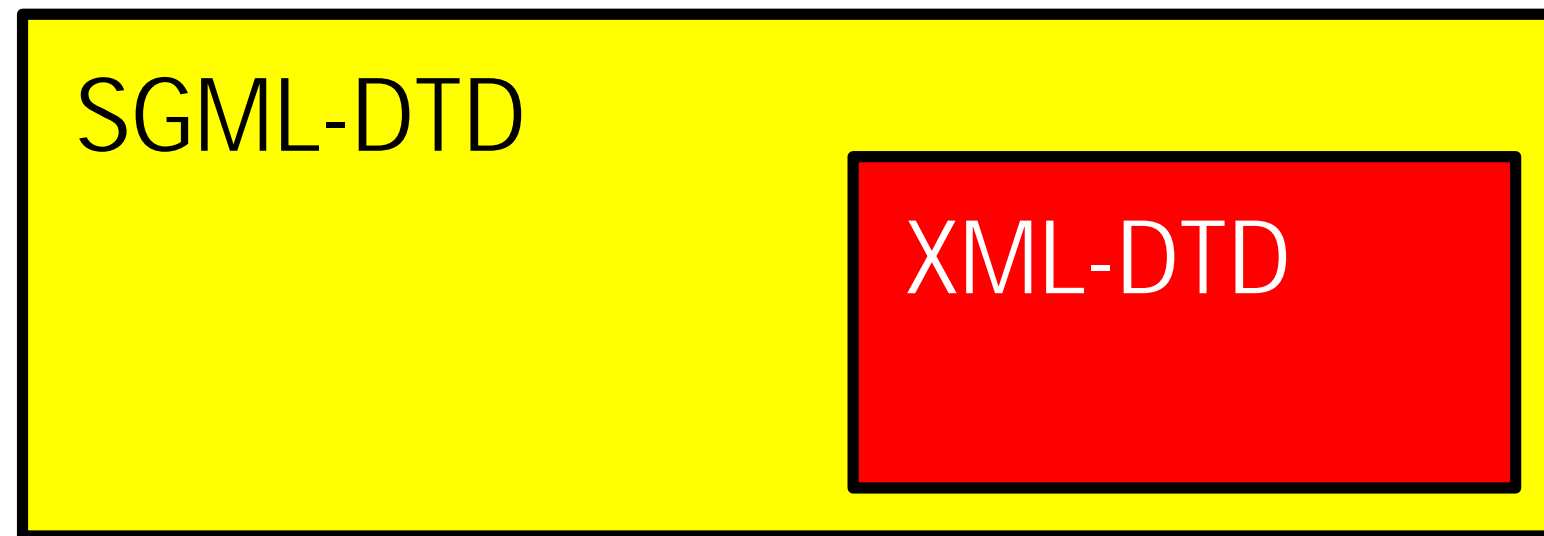
# Einführung und Motivation



## Einführung und Motivation

- DTD
  - ... definiert ein nicht-XML-Vokabular zur Strukturbeschreibung
  - ... offeriert ein geschlossenes eng-limitiertes Typsystem
- InfoSet
  - ... definiert ausschließlich Struktur
  - ... trifft keine Aussage über die syntaktischen Inhaltsdarstellung
  - ... gibt Bindung an (SGML-)Dokumenttypdefinition auf
- XML-Schema
  - Strukturelle Validierung
  - Typisierung
  - Erhalt und Erweiterung der DTD-Mächtigkeit

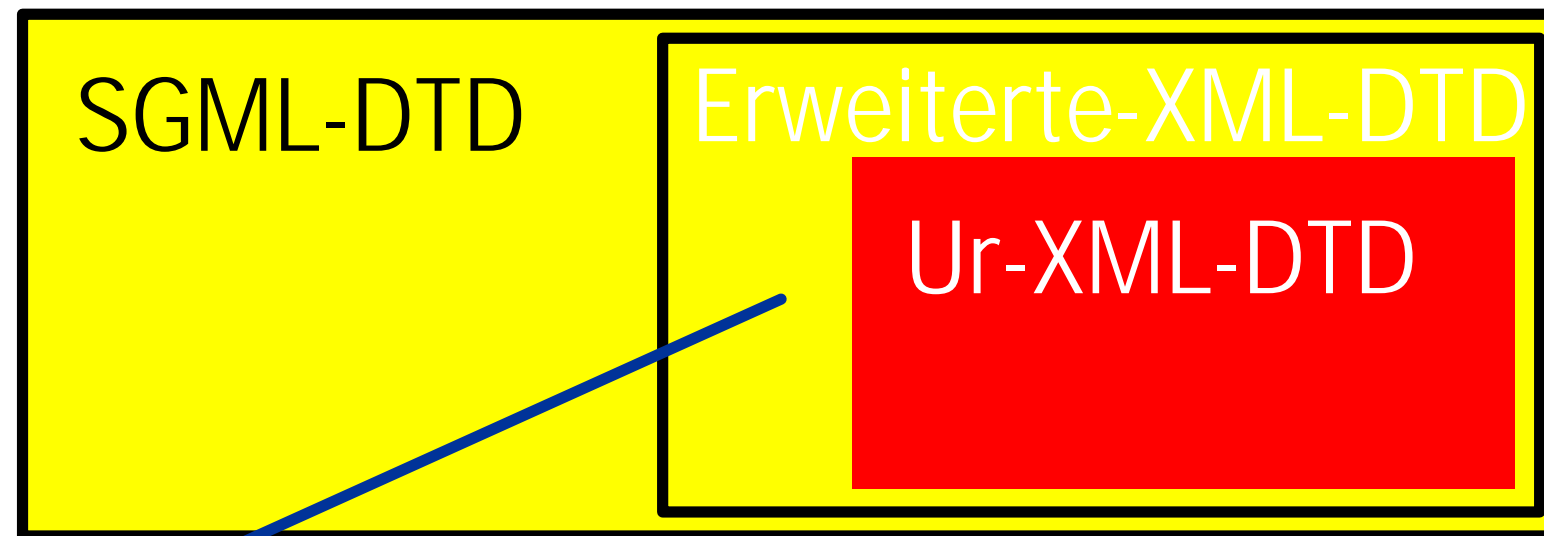
## XML-Schema: Verhältnis zur Document Type Definition



XML-DTD bildet syntaktisch und hinsichtlich der Mächtigkeit eine echte Untermenge des SGML-DTD-Mechanismus



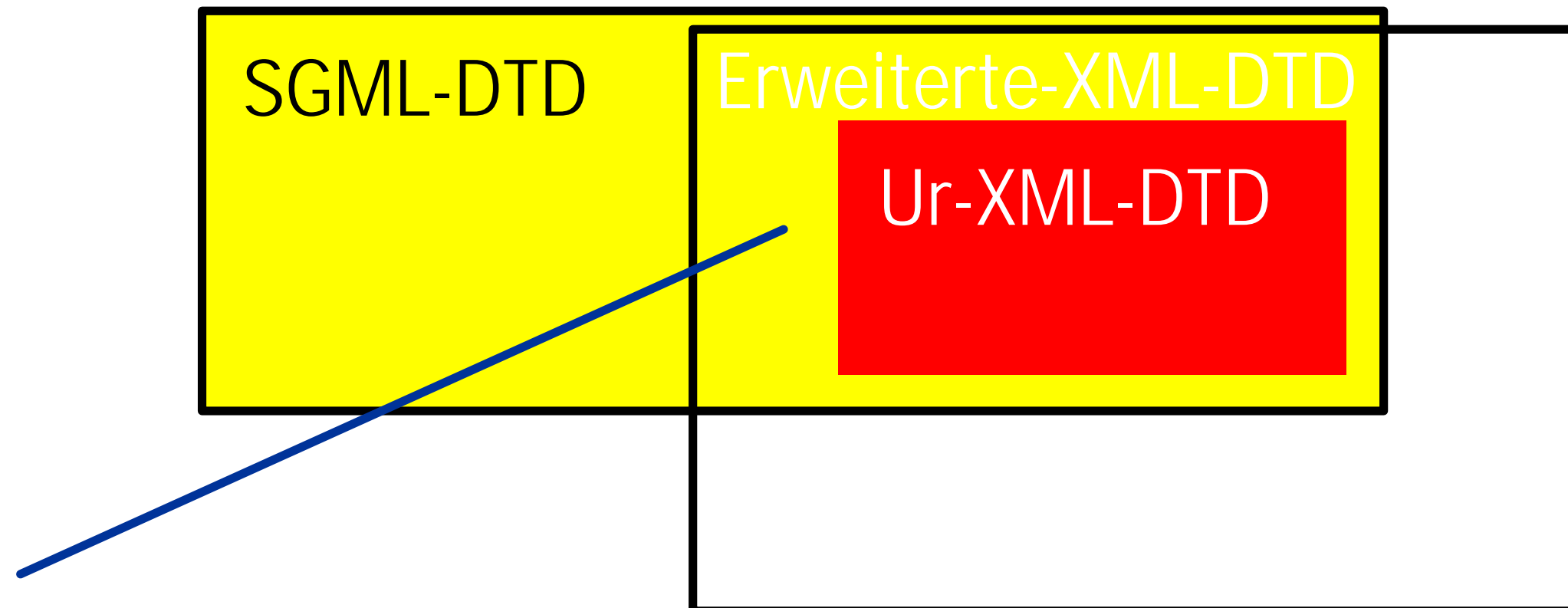
## XML-Schema: Erweiterungsoptionen



Erweiterung der XML-DTD-Mächtigkeit um weitere (bereits existierende) Primitive des SGML-DTD-Mechanismus:

- (+) (Unbestreitbarer) Mächtigkeitsgewinn
- (-) (unvermeidlicher) Komplexitätszuwachs
- (-) SGML-Mächtigkeit bildet natürliche Obergrenze

## XML-Schema: Erweiterungsoptionen



Erweiterung der XML-DTD-Mächtigkeit um Primitive die über den SGML-DTD-Mechanismus hinausgehen:

- (+) Freiheitsgrad hinsichtlich beliebiger Erweiterungen
- (-) XML-Grammatiken keine Untermenge von SGML mehr

## XML-Schema: Konkurrierende Erweiterungsmöglichkeiten

### Erweiterungen des bestehenden (SGML-/XML-)DTD-Mechanismus

- Data Types for DTD (DT4DTD)

### Wissensbeschreibung

- Document Content Description for XML (DCD)  
*(RDF basierte Weiterentwicklung von XML-Data)*

### Inspiziert durch XML-API-Entwicklung

- Schema for Object oriented XML (SOX)

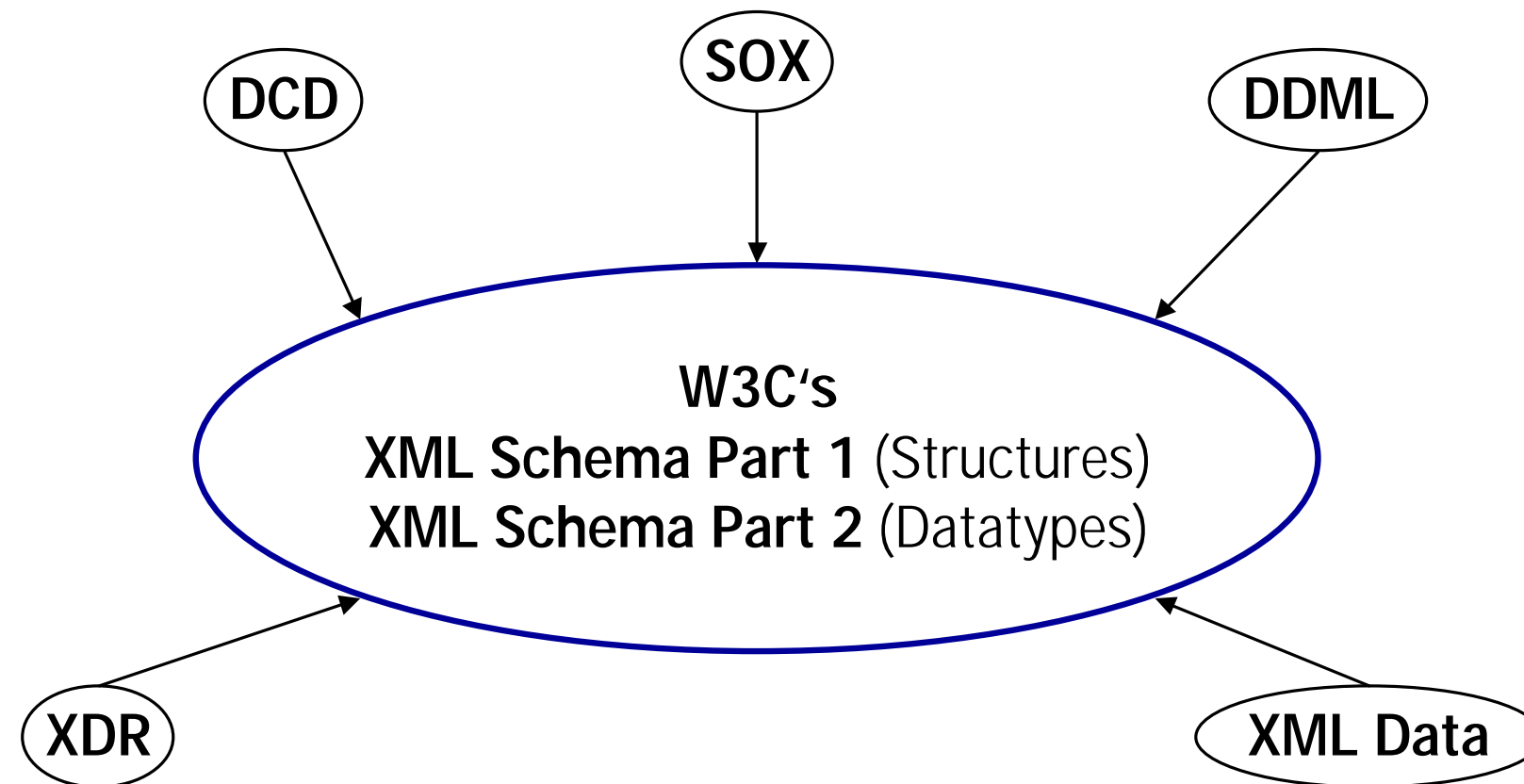
### XML-Sprachen zur Inhaltsbeschreibung

- Document Definition Markup Language/XSchema (DDML)
- Schematron (XSLT-basierte Auswertung der Dokumentstruktur)
- XML-Data/XML-Data Reduced (XDR) *(erster Ansatz noch vor Verabschiedung XML 1.0)*
- Document Structure Description (DSD)

## XML-Schema: Umfeld

- DTD
  - ... definiert ein nicht-XML-Vokabular zur Strukturbeschreibung
  - ... offeriert ein geschlossenes eng-limitiertes Typsystem
- InfoSet
  - ... definiert ausschließlich Struktur
  - ... trifft keine Aussage über die syntaktischen Inhaltsdarstellung
  - ... gibt Bindung an (SGML-)Dokumenttypdefinition auf
- XML-Schema
  - Strukturelle Validierung
  - Typsystem (erweitert und erweiterbar) und –zuweisung
  - Konzeptionelle Obermenge der DTD-Mächtigkeit
  - Einige konkurrierende Vorschläge und Konzepte
  - Bekanntester Ansatz: W3Cs XML-Schema

## XML-Schema: Vorgängeransätze

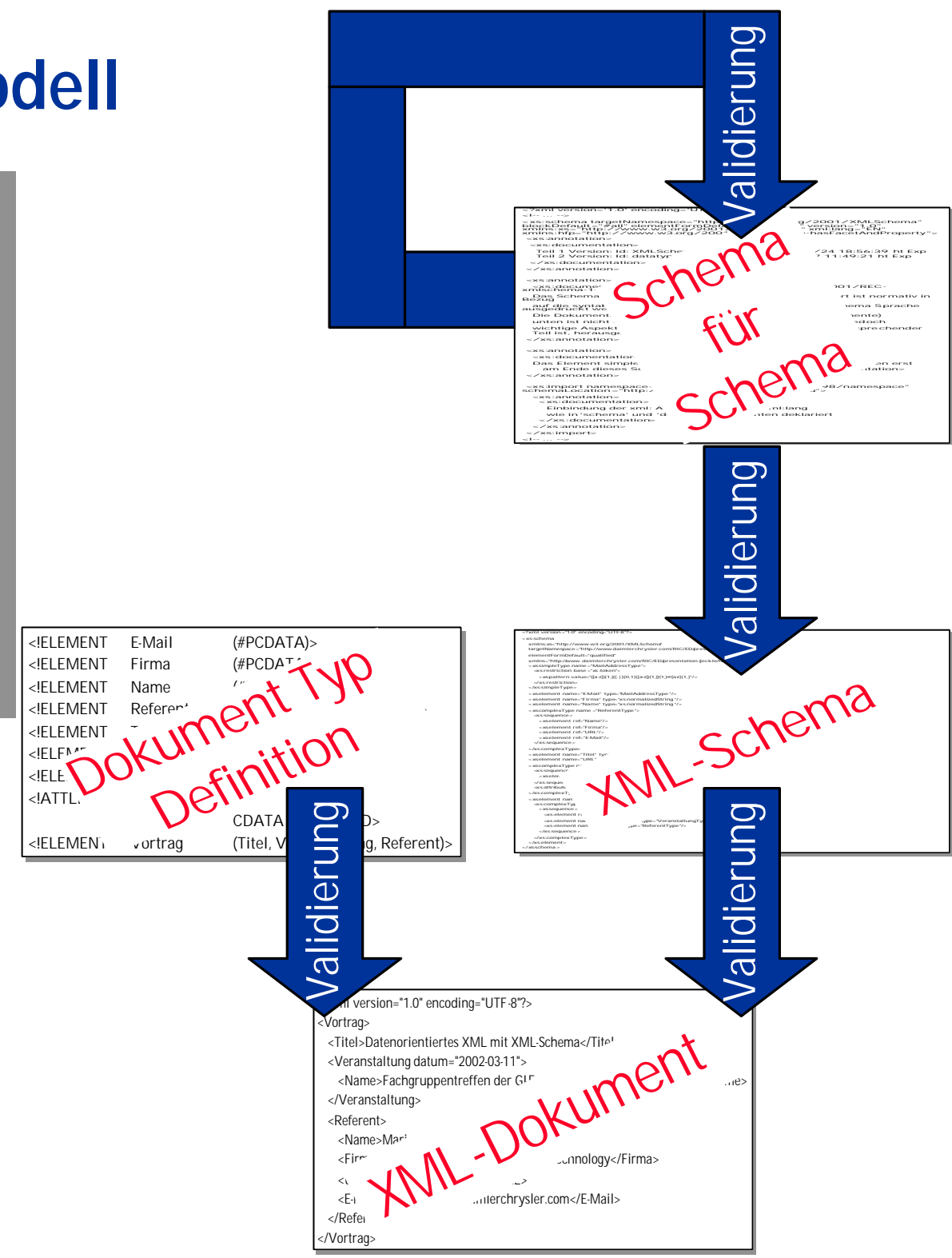


- XML-Schema
  - Strukturelle Validierung
  - Typsystem (erweitert und erweiterbar) und –zuweisung
  - Konzeptionelle Obermenge der DTD-Mächtigkeit
  - Einige konkurrierende Vorschläge und Konzepte
  - Bekanntester Ansatz: W3Cs XML-Schema

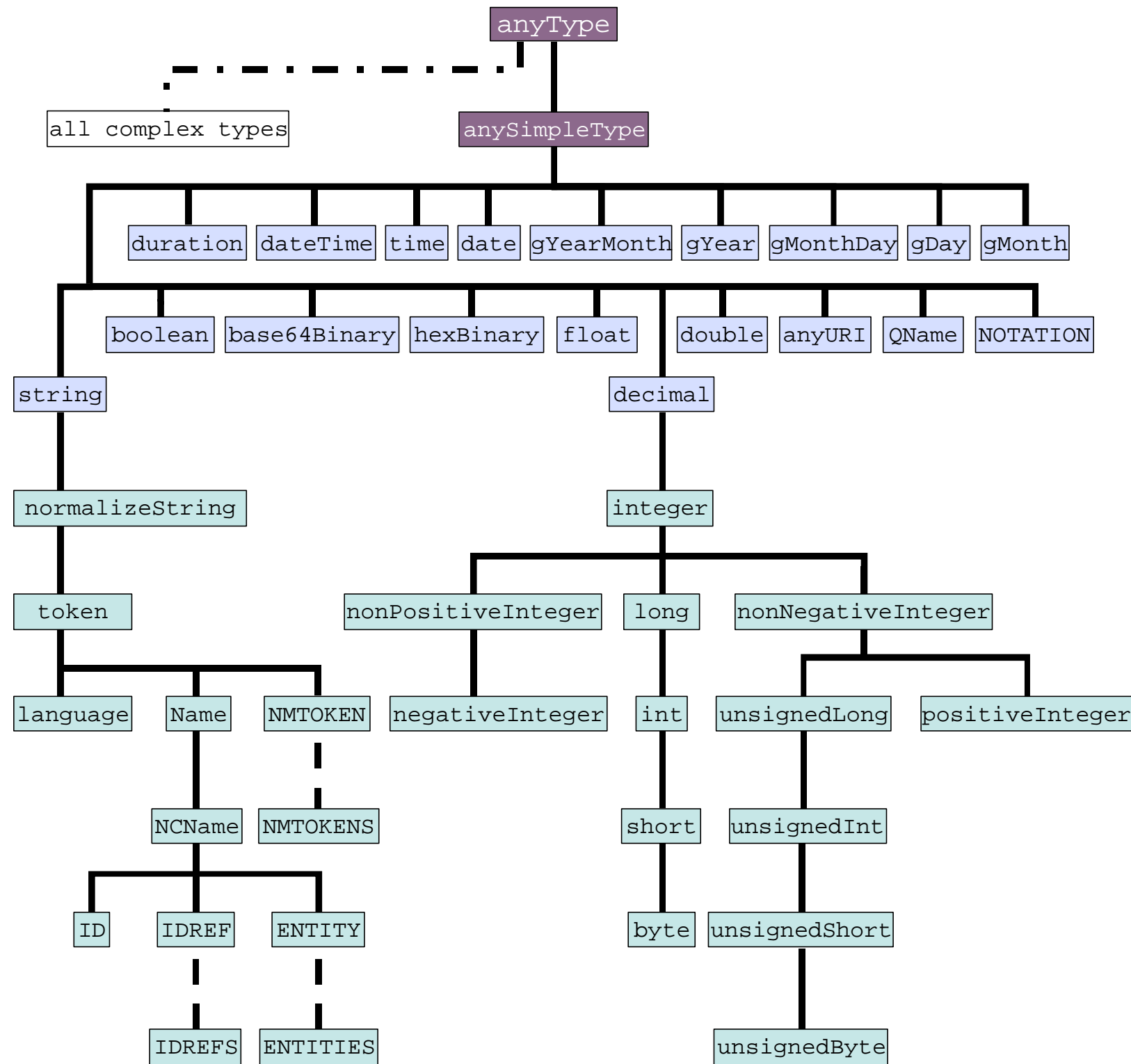
# XML-Schema: Organisation als Metamodell

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://example.com"
  xmlns:ex="http://example.com">
...
</xs:schema>
```

- Der XML-Schema-Standard
  - ... ist strukturell selbstbeschreibend
  - ... gestattet die Definition jedes Schemas als schema-gültiger XML-Datei
  - ... unterstützt XML-Namensräume (in zweifacher Hinsicht)
  - Besteht aus zwei Teilen:
    - Strukturen
    - Datentypen



# XML-Schema: Typsystem



- Urtyp
- Vordefinierter Primitivtyp
- Vordefinierter abgeleiteter Typ
- Komplexer Typ
- Abgeleitet durch Einschränkung
- - - - Abgeleitet durch Auflistung
- · - · - Abgeleitet durch Einschränkung oder Auflistung

# XML-Schema: Typsystem

positiveInteger

Abgeleitet

Primitiv

int

Atomar

Liste

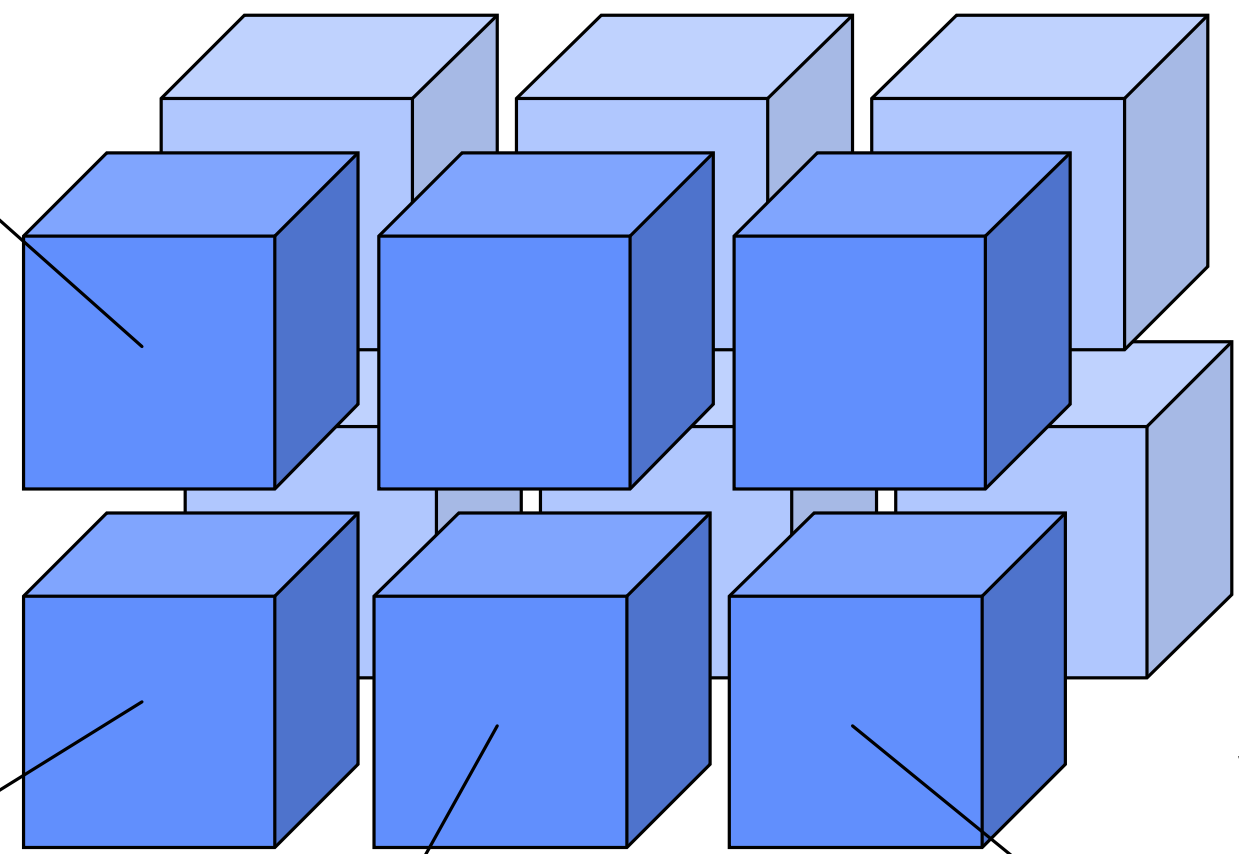
Vereinigung

namespaceList

Vordefiniert

Anwenderdefiniert

derivationSet





# XML-Schema: Typsystem

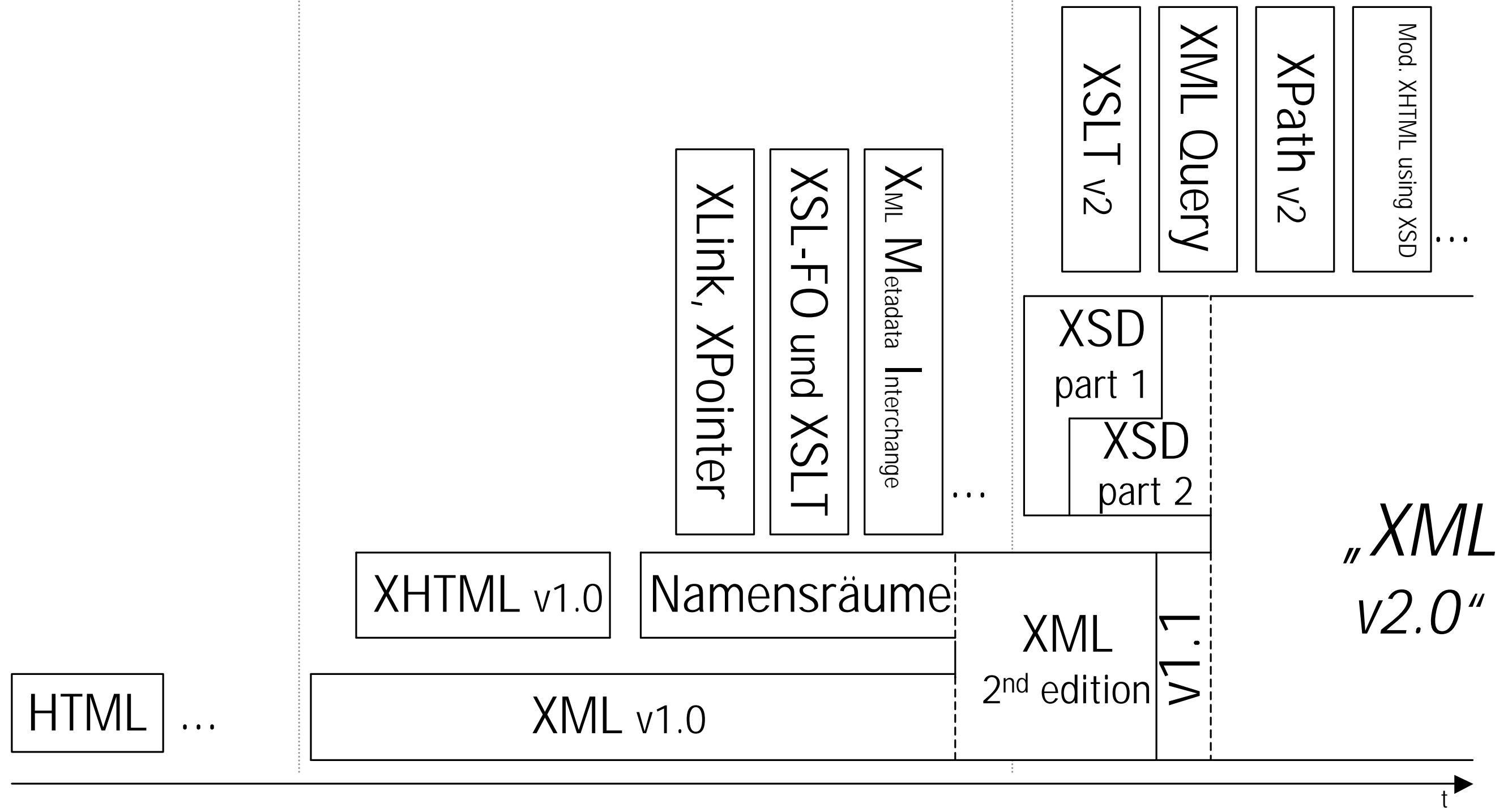
Typ	XML - Schema	Microsoft CLS	Java	SQL92/SQL3
$-2^7 \leq t \leq 2^7$	byte	-	byte	tinyint
$0 \leq t \leq 2^8 - 1$	unsignedByte	byte	-	
$-2^{15} \leq t \leq 2^{15} - 1$	short	short	short	smallint
$-2^{31} \leq t \leq 2^{31} - 1$	int	int	int	int, integer
$-2^{63} \leq t \leq 2^{63} - 1$	long	long	long	byte
Unicode-Zeichen(kette)	string	string	String	clob, char, character(n), varchar(n)
IEEE754	float double	float double	float double	float, real, double
{true,false,0,1}	boolean	bool	boolean	boolean
N	decimal	decimal	BigDecimal	numeric(p,s), dec, decimal(p,s)

# XML-Schema: Die XML-Sprachfamilie

Web-Prähistorie

1. Generation

2. Generation



## Anwendungen: Nullwerte

```
<... xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" ...>  
<xsi:nil="true" name="URL" xsi:nillable="true"/>
```

### XML-Instanzdokument

```
<... xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" ...>  
...  
<URL xsi:nil="true"/>  
...
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<Vortrag>  
  <Titel>Der XML-Schemaansatz als Voraussetzung  
    datenorientierter XML-Vokabulare der zweiten Generation</Titel>  
  <Veranstaltung datum="2002-06-18">  
    <Name>Informatik Kolloquium Uni Stuttgart</Name>  
  </Veranstaltung>  
  <Referent>  
    <Name>Mario Jeckle</Name>  
    <Firma>DaimlerChrysler Research and Technology</Firma>  
    <URL>http://www.jeckle.de</URL>  
    <E-Mail>mario.jeckle@daimlerchrysler.com</E-Mail>  
  </Referent>  
</Vortrag>
```

### NULL-Werte:

- Lukasiewicz-Logik für alle Skalartypen
- Analog: DBMS (SQL) und diverse Programmiersprachen

## Anwendungen: Anwenderdefinierte Bildung komplexer Typen

```
<xs:schema
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Vortrag">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Titel"/>
        <xs:element name="Veranstaltung">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Name"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<Vortrag>
  <Titel>Der XML-Schemaansatz als Voraussetzung
    datenorientierter XML-Vokabulare der zweiten Generation</Titel>
  <Veranstaltung datum="2002-06-18">
    <Name>Informatik Kolloquium Uni Stuttgart</Name>
  </Veranstaltung>
  <Referent>
    <Name>Mario Jeckle</Name>
    <Firma>DaimlerChrysler Research and Technology</Firma>
    <URL>http://www.jeckle.de</URL>
    <E-Mail>mario.jeckle@daimlerchrysler.com</E-Mail>
  </Referent>
</Vortrag>
```

### Vortrag

Titel

Veranstaltung

Name

## Anwendungen: Anwenderdefinierte Bildung komplexer Typen

```
<xs:schema
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Vortrag">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Titel"/>
        <xs:element name="Veranstaltung">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Name"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<Vortrag>
  <Titel>Der XML-Schemaansatz als Voraussetzung
    datenorientierter XML-Vokabulare der zweiten Generation</Titel>
  <Veranstaltung datum="2002-06-18">
    <Name>Informatik Kolloquium Uni Stuttgart</Name>
  </Veranstaltung>
  <Referent>
    <Name>Mario Jeckle</Name>
    <Firma>DaimlerChrysler Research and Technology</Firma>
    <URL>http://www.jeckle.de</URL>
    <E-Mail>mario.jeckle@daimlerchrysler.com</E-Mail>
  </Referent>
</Vortrag>
```

### Strukturierte Typen:

- Explizieren Hierarchisierung
- Basis der Wiederverwendung
- Analogon: Javas anonyme innere Klassen

## Anwendungen: Anwenderdefinierte Bildung skalarer Typen

Aggregation existierender Typen:

```
<xs:simpleType name="WarenkorbElemente">  
  <xs:list itemType="xs:token"/>  
</xs:simpleType>
```

Typkomposition:

```
<xsd:simpleType name="termin">  
  <xsd:union memberTypes="xs:date  
    NamenDerWochentage"/>  
</xsd:simpleType>
```

- Von existierenden Typen abhängige Neudefinition
- Implizite Substituierbarkeit gemäß XML-Schema-Typsystem bleibt erhalten
- Analog: Aggregation und Vereinigung in bekannten Programmiersprachen

## Anwendungen: Anwenderdefinierte Bildung skalarer Typen

### Vollständige Enumeration:

```
<xs:simpleType name="ampelfarben">
  <xs:restriction base="xs:string">
    <xs:enumeration value="rot"/> ...
  </xs:restriction>
</xs:simpleType>
```

### Domänenrestriktion:

```
<xsd:simpleType name="Postleitzahl">
  <xsd:restriction base="xs:string">
    <xsd:length value="5"/>
  </xsd:restriction>
</xsd:simpleType>
```

### Reguläre Ausdrücke:

```
<xs:simpleType
  name="MailAddressType">
  <xs:restriction base="xs:string">
    <xs:pattern
      value="([a-z]){1,}((.|-){0,1}([a-z]){1,}){1,}@([a-z]){1,}"/>
    </xs:restriction>
  </xs:simpleType>
```

- Beschränkung existierender Typen
  - ... hinsichtlich der zugelassenen Extensionen
  - ... hinsichtlich eines Merkmals
- Analog:
  - Domänenbildung in DBMS
  - Reguläre Ausdrücke (z.B. Perl)

## Kritik am Ansatz

- XML-Schema Syntax teilweise unnötig kompliziert
  - Dies kann ein Tribut an zukünftige Mächtigkeit sein
  - Heute: Allenfalls eingeschränkte Benutzbarkeit ohne Werkzeuge
- XML-Schema erweitert die DTD-Mächtigkeit teilweise nur (zu) behutsam
  - Strukturell stellt XML-Schema eine partielle Rekonstruktion der basalen DTD-Mächtigkeit dar.
  - Teilweise werden Anwendungsmöglichkeiten der DTD and Satellitenstandards delegiert (z.B. XML Include).
- XML-Schema ist ...
  - ... inzwischen weit verbreitet aber teilweise werden dennoch Alternativansätze (z. B. *RELAX NG* (ISO)) weiter vorangetrieben.
  - ... nur der erste Schritt, weitere werden folgen.



# Zukünftige Entwicklungen

---

## XML Schema 1.1 You can help

---

The XML Schema WG is currently working to develop a set of requirements for XML Schema 1.1, which is intended to be mostly compatible with XML Schema 1.0 and to have approximately the same scope, but also to fix bugs and make whatever improvements we can, consistent with the constraints on scope and compatibility.

If you have suggestions for specific requirements for XML Schema 1.1 (or for later versions), please send us mail using the [public comment list](#). Please note that the comments list is not only public-write but also public-read; don't say things you don't want seen in public.

Thanks!

## Zukünftige Entwicklungen

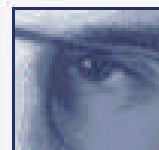
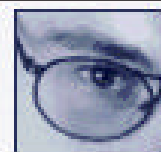
- XML-Schema Version 1.1:
  - Fortschreibung des *Primer*-Dokuments und Synchronisation mit neuen Sprachmitteln aus Part 1 und Part 2
  - Erhöhung der Lesbarkeit der normativen Dokumente (Part 1 und Part 2)
  - Behebung der bekannten Fehler (Errata-Dokument)
  - Behutsame Erweiterungen
  - Insgesamt: Bereinigter Interimsstand
- XML-Schema Version 2.0:
  - Klare Trennung zwischen Informationsmodell und Validierungsanteilen

## Zukünftige Entwicklungen: XML-Schema v1.1

- Datentypen: Schreibungssensitivität birgt Probleme im Zusammenspiel mit Programmiersprachen.
- Strukturen: Transitive Auswertung freier Inhaltsmodelle (sog. *wildcard transitive bug*).
- Konsistenzgarantierende Einschränkungen auf Attributebene.
- Verkettete referenzierende Einschränkungen.
- Internationalisierungsaspekte.
- Überarbeitung der einschränkenden Ableitung.
- Mehrfachvererbung.
- Modularisierung und Untermengenbildung.

## Zukünftige Entwicklungen: XML-Schema v2.1

- Verkettete referenzierende Einschränkungen.
- Mehrfachvererbung.
- Interaktion verschiedener Einschränkungen bei Substitution.
  
- Reguläre Ausdrücke zur Element- und Attributbenennung.
- Mächtig(er)e konsistenzgarantierende Einschränkungen.
- Inhaltsbasierte Einschränkungen (sog. *Co-Constraints*).
- ...



[Mario Jeckle ...](#)  
[Dialog ...](#)  
[über diese Seiten ...](#)  
[suchen ...](#)  
[SiteMap](#)

[Unified Modeling Language \(UML\)](#)  
[eXtensible Markup Language \(XML\)](#)  
[XML Metadata Interchange \(XMI\)](#)  
[Web Services](#)  
[XML Acronym Demystifier Project](#)

[Vorträge und Publikationen](#)  
[Vorlesungen](#)  
[Diplomarbeiten](#)  
[GOOAL.net](#)  
[XML-Arbeitskreis](#)

[Web Services Workshop \*WS-RSD'02\*](#)  
[Internet > Search Engines](#)  
[Mersennesche Primzahlen](#)  
[Feedback](#)  
[Rotkreuz Mitgliederverwaltung](#)

# jeckle.de

Including ...

- [German Translation of XML Schema](#)
- [XML Acronym Demystifier](#)
- [Lecture Notes on XML Schema](#)