

# Semistrukturierte Daten

## Namespaces

Stefan Woltran  
Emanuel Sallinger

Institut für Informationssysteme  
Technische Universität Wien

Sommersemester 2014

# Namespaces

Namespaces (offiziell *Namespaces in XML*) werden benötigt wenn

- Element- und Attributnamen **mehrerer** Standards in **einem** Dokument vorkommen sollen
- Anwendungen **erkennen** müssen, welche Elemente und Attribute für die Verarbeitung durch sie bestimmt sind

Namespaces sind nicht offizieller Teil der XML (1.0) Spezifikation, aber

- fast alle auf XML aufbauende Standards verwenden Namespaces
- es ist stärkstens zu empfehlen, keine XML Dokumente zu produzieren, die gegen die Namespaces Spezifikation verstossen

# Beispiel (Problem)

Bewertung der LVA durch Studierende:

```
<veranstaltung>  
  <titel>Semistrukturierte Daten</titel>  
  <bewertung>naja...</bewertung>  
</veranstaltung>
```

Bewertung der LVA durch das Vorlesungsverzeichnis:

```
<veranstaltung>  
  <titel>Semistrukturierte Daten</titel>  
  <bewertung>LVA eines Wahlmoduls</bewertung>  
</veranstaltung>
```

# Beispiel (Lösung)

```
<veranstaltung  
  xmlns:stud="http://www.oeh.ac.at/bew/"  
  xmlns:vorl="http://tiss.tuwien.ac.at/lehre/3.0">  
  <titel>Semistrukturierte Daten</titel>  
  <stud:bewertung>naja...</stud:bewertung>  
  <vorl:bewertung>LVA eines Wahlmoduls</vorl:bewertung>  
</veranstaltung>
```

# Namespaces

Als **Namespace Namen** werden URIs verwendet

- zwei Namespace Namen sind gleich, wenn sie als Strings gleich sind
- die URI braucht keine URL zu sein
- auch der leere String ist als Namespace Name zulässig

Beispiele für Namespace Namen:

- `http://www.w3.org/1999/xhtml` (XHTML)
- `http://www.w3.org/1999/XSL/Transform` (XSLT)

# Namespace Deklarationen

Eine **Namespace Deklaration** erfolgt durch ein Attribut der Form

```
xmlns:präfix="name"
```

Sie besteht aus

- Namespace Präfix: ein Name (nach den Regeln der XML Spezifikation)
- Namespace Name: eine URI

Beispiel für eine Namespace Deklaration:

```
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
```

# Qualifizierte Namen

Für Elemente und Attribute werden **qualifizierte Namen** verwendet

```
prefix:local-name
```

Sie bestehen aus

- Namespace Präfix: ein Name
- Lokaler Name: ein Name

Beispiel für qualifizierte Namen:

```
xsl:transform
```

# Beispiel (Default Namespace)

```
<veranstaltung  
  xmlns:stud="http://www.oeh.ac.at/bew/"  
  xmlns="http://tiss.tuwien.ac.at/lehre/3.0">  
  <titel>Semistrukturierte Daten</titel>  
  <stud:bewertung>naja...</stud:bewertung>  
  <bewertung>LVA eines Wahlmoduls</bewertung>  
</veranstaltung>
```



# Namespace Deklarationen

Eine **Default Namespace Deklaration** erfolgt durch ein Attribut der Form

```
xmlns="name"
```

Sie besteht nur aus

- Namespace Name: eine URI

Beispiel für eine Default Namespace Deklaration:

```
xmlns="http://www.w3.org/1999/XSL/Transform"
```

# Qualifizierte Namen

Ein **qualifizierter Name** kann auch kein Präfix verwenden

`local-name`

Er besteht dann nur aus

- Lokaler Name: ein Name

Als Namespace Name wird verwendet:

- für **Elemente**: der Default Namespace
- für **Attribute**: der leere Namespace

# Beispiel (Überschreiben)

```
<veranstaltung  
  xmlns="http://tiss.tuwien.ac.at/lehre/3.0">  
  <titel>Semistrukturierte Daten</titel>  
  <bewertung xmlns="http://www.oeh.ac.at/bew/">  
    naja...  
  </bewertung>  
  <bewertung>LVA eines Wahlmoduls</bewertung>  
</veranstaltung>
```

# Überschreiben von Namespaces

Ein Präfix oder der Default Namespace

- können in einem Dokument mehrfach deklariert werden
- es gilt immer die **näheste** Deklaration entlang der Vorfahren:
- zuerst eigenes Element, dann Elternelement, usw. bis zum Wurzelement

Wenn es keine Deklaration entlang der Vorfahren gibt

- für den **Default Namespace**: wird der leere Namespace verwendet
- für ein **Präfix**: ist das ein Fehler (wenn das Präfix verwendet wird)

# Qualifizierte und Expandierte Namen

Der **qualifizierte Name** eines Elements oder Attributs besteht aus

- Namespace **Präfix** (optional)
- Lokaler Name

Der **expandierte Name** eines Elements oder Attributs besteht aus

- Namespace **Name**
- Lokaler Name

Kein Element darf zwei Attribute mit gleichem **expandierten** Namen enthalten!