





# Virtuelle Realität

## X3D / VRML



G. Zachmann  
 Clausthal University, Germany  
[cg.in.tu-clausthal.de](http://cg.in.tu-clausthal.de)



## X3D / VRML

- Was ist X3D/VRML?
  - Scenegrph & File-Format, plus ...
  - Multimedia-Support
  - Hyperlinks
  - Verhalten und Animationen
- Achtung: VRML  $\neq$  VR !
- Varianten:
  - VRML 1.0 (1995) (= Inventor, also kein VRML)
  - VRML 2.0 (1996)
  - VRML97 (1997) – ISO Standard, praktisch identisch zu VRML2
  - X3D (2003): ISO Standard, im wesentlichen andere Syntax, nämlich XML

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 2

## Vorteile von X3D

- Die Spezifikation von VRML ist an einigen Stellen nicht eindeutig
  - In X3D präzisiert
- X3D hat 100+ Knoten (aufgeteilt in Components / Profiles)
  - VRML hat nur 54 Knoten
- X3D hat 3 verschiedene sog. "File Encodings":
  - **Classic**: sieht aus wie VRML; Suffix = `.wrl` oder `.x3dv`
    - Jede Software, die X3D lesen kann, kann (im Prinzip) auch VRML lesen
  - **XML**; Suffix = `.x3d`
    - das ist das Format, das man i.A. unter "X3D" versteht
  - **Binary** (XML braucht sehr viel Platz); Suffix = `.x3db`
    - Trick, falls man einen binären VRML-File editieren möchte: einfach xxx.wrl umbenennen nach xxx.wrl.gz, dann mit 'gunzip xxx.wrl.gz' auspacken ;-)

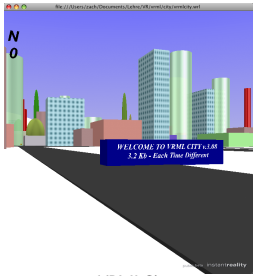
G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 3

## X3D-"Browser"

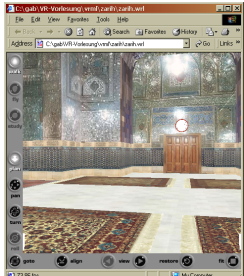
- InstantReality ([www.instantreality.org](http://www.instantreality.org)):
  - Läuft auf allen 3 Plattformen
  - Implementiert (angeblich) V3.1 von X3D komplett
- FreeWrl ([freewrl.sourceforge.net](http://freewrl.sourceforge.net)):
  - Läuft auf Linux & Mac OS X
  - Implementiert das Subset (Profile) "Interchange" von X3D
- Octaga ([www.octaga.com](http://www.octaga.com)):
  - Windows & Mac OS X
- BS Contact ([www.bitmanagement.com](http://www.bitmanagement.com)):
  - Nur Windows ;-)
- Cortona (<http://www.parallelgraphics.com/products/cortona/>):
  - Nur als Plugin für Web-Browser
- XVD (<http://www.vrmedia.it/Xvr.htm>):
  - Nur Internet Explorer unter Windows ☹

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 4


## Beispiele



VRMLCity



Zarih



HSV-Arena ([http://www.hsv-hshnordbank-arena.de/de/die\\_arena/die\\_arena\\_in\\_3d.html](http://www.hsv-hshnordbank-arena.de/de/die_arena/die_arena_in_3d.html))  
(leider nur mit proprietärem Plugin)

G. Zachmann Virtuelle Realität und Simulation - WS 09/10
X3D / VRML97 5

## Literatur, References

- Bücher:
  - Don Brutzman, Leonard Daly:  
*X3D: Extensible 3D Graphics or Web Authors*. Morgan Kaufman, 2007.
  - Andrea L. Ames, David R. Nadeau, and John L. Moreland:  
*The VRML 2.0 Sourcebook*. John Wiley & Sons, 1996.
  - Hartman, Jed, and Wernecke:  
*The VRML 2.0 Handbook*. Addison-Wesley, 1996.
- Online: Auf der Homepage zur Vorlesung
  - The Annotated VRML97 Reference
  - Der X3D-Standard: Knoten, Javascript, Java
- Die online Doku zu InstantReality:
  - Tutorials
  - Übersicht aller Knoten

G. Zachmann Virtuelle Realität und Simulation - WS 09/10
X3D / VRML97 6

## Web-Seiten

- Die Web-Seite zum X3D-Buch:  
[www.x3dgraphics.com](http://www.x3dgraphics.com)  
mit Bspielen, Tools, ...
- Eine "Meta"-Seite beim Web3D-Konsortium:  
[www.web3d.org/x3d/content/examples/X3dResources.html](http://www.web3d.org/x3d/content/examples/X3dResources.html)  
mit Links zu Software für Viewer, Konverter, Authoring-Tools, Plugins, Beispielen, Büchern, etc.

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 7

## Encodings am Bsp. der trivialen X3D-Szene

- Als X3D-Encoding:
 

```

<?xml version="1.0" encoding="UTF-8"?>
<X3D profile='Full'>
<Scene>
<!-- empty scene -->
</Scene>
</X3D>

```

Annotations:

  - XML file declaration
  - Scene-Tag, entspricht Wurzel-Kn.
  - X3D-Tag, geklammert, analog zum <html>-Tag in HTML
  - a comment
- Als ClassicVRML-Encoding:
 

```

#X3D V3.1 utf8
PROFILE Full
# empty scene

```
- und als VRML97:
 

```

#VRML V2.0 utf8
# empty scene

```

- Der Wurzel-Knoten für die gesamte Szene ist in VRML implizit!
- Keine Profiles und viel weniger Knoten in VRML97

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 8

- Gründe für das XML-Encoding:
  - Ähnlichkeit zu HTML (Tags und Attribute: `<tag attr="val">...</tag>`)
  - XML ist ASCII (wie VRML97), also im Prinzip "human readable" (im Gegensatz zu binären Formaten)
  - XML ist ein weit verbreiteter Standard zur Beschreibung von Daten
  - XML ist eine Familie von Technologien: CSS, XSLT, Xpointer, ...
  - XML ist Lizenz-frei
- Gründe für das ClassicVRML-Encoding:
  - Legacy-Daten ("Altlasten")
  - Für Menschen leichter zu lesen und zu schreiben

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 9

- Der X3D-"Browser" (stand-alone oder embedded):

The diagram illustrates the internal structure of an X3D Browser. It is divided into several functional layers:


- External Interactions:**
  - Top left: X3D scenes, X3D streams (input/output).
  - Top right: Event passing within HTML web page or external application (input/output).
- Core Browser Components:**
  - Parsers:** X3D XML encoding, Classic VRML encoding, Binary encoding.
  - Scene Authoring Interface (SAI):** Application programmer interfaces.
  - New node and prototype construction:** X3D nodes, node types; Prototype and External Prototype.
  - Scripting engines:** EcmaScript, Java, others.
  - Scene graph manager:** Manages the scene graph.
  - Scene Graph Renderable Nodes:** The visual representation of the scene.
  - Event Graph Animation Nodes:** Handle animation and events.

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 10

## Hello World

- In X3D (genauer: XML-Encoding):
 

```
<?xml version="1.0" encoding="UTF-8"?>
<X3D profile='Immersive'>
<Scene>
  <Shape>
    <Text string="Hello" "world!" />
  </Shape>
</Scene>
</X3D>
```


- In VRML:
 

```
#X3D V3.1 utf8
Shape {
  geometry Text {
    string [ "Hello" "world!" ]
  }
}
```

Tip: ASCII-Editor verwenden, der matching brackets erkennt und als Texteinheit behandeln kann

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 11

## Knoten und Felder

- Knoten dienen zur Beschreibung ...
  - ... des **Szenengraphen** (die üblichen Verdächtigen):
    - Geometry, Transform, Group, Lights, LODs, ...
  - ... des **Verhaltensgraphen** (*behavior graph*), d.h., des Verhaltens der Objekte und bei User-Input
- Knoten := Menge von Feldern
  - "Single-valued Fields" und "Multiple-valued Fields"
  - Jedes Feld eines Knotens hat einen eindeutigen Namen
  - Diese Namen sind per Spezifikation vordefiniert
- Feldarten:
  - field** = Daten im File
  - eventIn**, **eventOut** = s.u., werden nicht gespeichert
  - exposedField** = Kombination der drei (xxx, set\_xxx, xxx\_changed)

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 12

## Feld-Typen

- Alle Feldtypen gibt es als "single valued"- (**SF**...) und als "multiple valued"-Variante (**MF**...)
- Beispiel für ein SFField:
 

```
<Material diffuseColor="0.1 0.5 1" />
```

 X3D
 

```
material Material {
  diffuseColor 0.1 0.5 1
}
```

 VRML
- MFField's sind im Prinzip nichts anderes als **Arrays**
  - Falls der Grundtyp ein **Tuple** ist (z.B. Farbe oder Vektor), sollte man in X3D die einzelnen Elemente mit Komma trennen. Beispiel:
 

```
"1 0 0, 0 1 0, 0 0 1"
```
  - In VRML müssen MFField's mit [] geschrieben werden. Beispiel:
 

```
[ 1 0 0, 0 1 0, 0 0 1 ]
```

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 13

- Grundtypen: die üblichen Verdächtigen:
 

Field type	X3D example	VRML example
SFBool	true / false	TRUE / FALSE
SFInt32	12	-17
SFFloat	1.2	-1.7
SFDouble	3.1415926535	
SFString	"hello"	"world"

Erinnerung: zu jedem SF-Feld gibt es ein MF-Feld
- Etwas höhere Datentypen:
 

Field type	Beispiel
SFColor	0 0.5 1.0
SFColorRGBA	0 0.5 1.0 0.75
SFVec3f	1.2 3.4 5.6
SFMatrix3f	1 0 0 0 1 0 0 0 1
SFString	"hello"

Anmerkungen:

  - Die Werte in SFColor müssen in [0,1] liegen
  - Analog gibt es die Varianten \*2f, \*3f und \*4f.

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 14

Field type	X3D example	VRML example
<b>SFImage</b>	enthält spezielle Pixel-Encodings	
<b>SFNode</b>	<code>&lt;Shape&gt; ... &lt;/Shape&gt;</code>	<code>Shape { ... }</code>
<b>MFNode</b>	<code>&lt;Shape&gt;... , &lt;Group&gt;...</code> oder <code>&lt;Transform&gt;...</code>	<code>Transform {</code> <code>  children [... ] }</code>
<b>SFRotation</b>	<code>0 1 0 3.1415</code>	
<b>SFTime</b>	<code>0</code>	

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 15

<ul style="list-style-type: none"> <li>▪ <b>Anmerkungen zu SFImage:</b> <ul style="list-style-type: none"> <li>▪ Der Wert des Feldes ist eine Folge von Zahlen: Breite, Höhe, Anzahl Komponenten pro Pixel, Pixel, Pixel, ...</li> <li>▪ Pro Kanal Werte im Bereich [0,255]</li> <li>▪ Beispiel (bei 3 Komponenten): <code>0xFF0000</code> = Rot, <code>0x00FF00</code> = Grün, ...</li> <li>▪ Beispiel für ein vollständiges <b>SFImage</b>:           <pre>2 4 3 0xFF0000 0xFF00 0 0 0 0 0xFFFFF0 0xFFFF00 # w h c red green black.. white yellow</pre> </li> </ul> </li> <li>▪ <b>SFImage</b> ist nur für sehr kleine Texturen gedacht und kommt nur im Knoten <b>PixelTexture</b> vor           <ul style="list-style-type: none"> <li>- Hintergrund: man wollte eine Möglichkeit haben, Texturen algorithmisch zu erzeugen (mittels Java)</li> </ul> </li> <li>▪ Für große ("richtige") Texturen verwende man PNGs oder JPGs und den Knoten <b>ImageTexture</b></li> </ul>
---

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 16



- Generelle Anmerkungen zum Design:
  - Das Design ist insofern **orthogonal**, als es zu jedem **SF**-Typ einen **MF**-Typ gibt
  - Das Design ist insofern **nicht orthogonal**, als manche Typen generisch sind (z.B. **SFBool**, **SFVec3f**), andere wiederum eine festgelegte Semantik haben (z.B. **SFColor**, **SFTime**, etc.)
    - Es ist nicht ganz klar, ob dies gut/schlecht ist ...

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 17

## Die Spezifikation der Knoten

- Knoten werden definiert durch ihre Felder und deren Bedeutung
- Die Syntax zur Definition von Knoten (vorerst):
 

```
Name_of_Node_Class {
    type_of_field name_of_field_1 default_value
    type_of_field name_of_field_2 default_value
    ...
}
```
- Bemerkungen:
  - Die Defaults werde ich im Folgenden meist weglassen
  - Auch werde ich nicht alle Felder aufzählen, nur die wichtigsten
  - Im folgenden werden nur einige wenige Knoten besprochen
- Fazit: schauen Sie in die Doku und das Tutorial!

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 18

- Beispiel:

```
Cone {  
  SFFloat  bottomRadius 1  
  SFFloat  height      2  
  SFBool   side        TRUE  
  SFBool   bottom      TRUE  
}
```
- Verwendung:

```
Cone { bottomRadius 1 height 2 }
```

 VRML-Syntax

```
<Cone bottomRadius="1" height="2" />
```

 XML-Syntax
- Bemerkung: Cone ist in XML-Syntax ein sog. **Singleton-Element**, d.h., es gibt kein öffnendes/schließendes Tag-Paar!

G. Zachmann Virtuelle Realität und Simulation - WS 09/10 X3D / VRML97 19