3D- Produktpräsentationen im Internet

Prof. Dr.-Ing. Ralf Böse

Fachhochschule Schmalkalden
FB Informatik

Abstract: In einem an der Fachhochschule bearbeiteten Thema zum digitalen Produktdaten Management wurden neben Schnittstellenuntersuchungen zwischen den einzelnen Prozessschritten auch die Möglichkeiten zur interaktiven 3D Produktpräsentation im Internet untersucht. Der Beitrag beginnt mit einem kurzen Überblick zur Geschichte von 3D Computergrafiken im Internet. Aufbauend auf einer Diskussion des klassisches Standard VRML 97 werden die neuen Ansätze des W3C Konsortiums mit X3D und konkurrierende Entwicklungen, hier Viewpoint Experience Technology VET, diskutiert. Abschließend werden einige praktische Projektergebnisse vorgestellt.

1 Motivation

In der heutigen betrieblichen Praxis, insbesondere bei komplexen Systemherstellern und ihren Zulieferern, ist die Applikation eines durchgehenden digitalen Produktdatenmanagements zur Überlebensfrage im internationalen Wettbewerb geworden. Diese Problematik wurde in einem fachgebietsübergreifenden Projekt zwischen dem Maschinenbau und der Informatik, welches vom BMBF im Programm aFuE an Fachhochschulen (ProDat [13], FKZ: 1705799) gefördert wurde, vor dem Hintergrund konkreter Kooperationsprojekte untersucht.
Die nachfolgende Grafik stellt den digitalen Workflow dar und verdeutlicht auch den Zusammenhang mit modernen Simulationstechniken, wie VR- Anwendungen auf der Basis eines digitalen Modells. Dieses Modell kann bereits vor der Einführung eines Produktes für die Werbung, Schulung und Dokumentation über das Internet oder über andere Medien genutzt werde, um die Time to Market Zeit zu verkürzen.

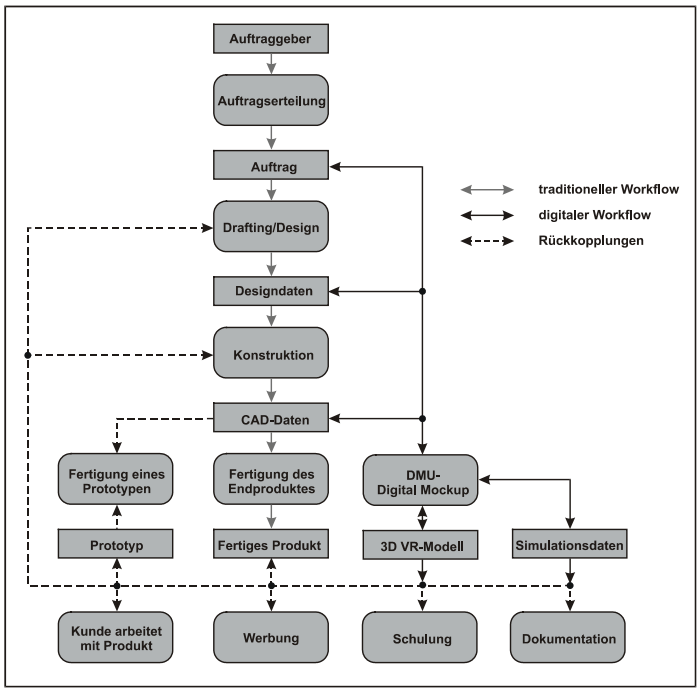


Abbildung 1: Der vereinfachte Workflow eines Produktes

2 3D-Internetdarstellungsformate

2.1 Stand der Technik

Probleme bereiten auch heute noch die unzureichenden Technologien zur interaktiven und echtzeitfähigen Präsentation von 3D Daten über das Internet. Einige seien kurz aufgeführt:

- Mangelnde Bandbreiten,
- veraltete Standards,
- intransparente Viewer/ Browse Technologien,
- schlechte Integration in allgemeine Internet Technologien,
- fehlende Entwicklungswerzeuge und
- keine anerkannten, standardisierten User Interface Technologien zur 3D Interaktion und für 3D Viewer

Andererseits gibt es für das Handling und die Visualisierung von 3D Geometrien und Topologien für arbeitsplatzgebundene Anwendungen eine lange erfolgreiche Tradition. Erinnert sei an die internationalen Standards GKS [1][2] und PHIGS [3],

die Meilensteine in der Entwicklung der Grafik darstellten. Vor dem Hintergrund heutiger komplexer, dezentraler Anforderungsprofile reichen diese Ansätze aber nicht mehr aus.
Lösungen zeichnen sich bei Visualisierungen über das Internet, trotz der großen Datenmengen, durch die Weiterentwicklungen auf folgenden Gebieten ab:

- Die Leistungsfähigkeit der am Arbeitsplatz verfügbaren Hardware ist drastisch im Bereich der 3D Darstellung gestiegen,
- Streaming- Technologien sind vor dem Hintergrund wachsender Bandbreiten verfügbar,
- Basis Technologie, wie OpenGL [9] und VRML [4], sind vorhanden bzw. werden weiterentwickelt,
- neue Entwicklungen im WEB Konsortium [22] mit XML [8] und X3D [22] können zu interrational anerkannten Standards werden und
- industrielle Entwicklungen sind schon am Markt, wie VET [27].

2.2 Anforderungen und Klassifikation

Die aufgeführten Anforderungen und Klassifikationen sind natürlich nicht umfassend und geben nur einen kleinen Ausschnitt wieder. Sie sollen aber das Verständnis für die im weiteren besprochenen internationalen Entwicklungen vertiefen.
Für die Präsentation dreidimensionaler Daten im Internet existieren viele Formate, von denen aber nur wenige standardisiert sind. Einige besitzen gewissermaßen den Status eines Standards, weil sie weit verbreitet und in eine Vielzahl von Programmen integriert worden sind. Außerdem existieren Formate, die z.B. aus Forschungs- oder anderen Spezialsystemen abgeleitet wurden, um 3D-Informationen über das Internet zu verbreiten. Einen Überblick gibt [16].
Ein wesentliches Kriterium bei der Bewertung von Formaten ist die freie Verfügbarkeit von 3D User Interface Technologien (Viewer oder Browser) und von Entwicklungswerkzeugen auf den wichtigsten Plattformen.
Dabei kommt der transparenten Integration in gängige Internet Technologien eine immer größere Bedeutung zu, um Datenbestände, z.B. in Datenbanken, konsistent verwalten zu können. Dies ist eine Voraussetzung, um moderne eCollaboration-Konzepte durchsetzen zu können. Der Datenaustausch auf der Basis XML (Extensible Markup Language) [8] ermöglicht dabei das Zusammenführen von Daten unterschiedlicher Quellen.
Ein anderer Aspekt ist die Standardisierung der Datenbasis. Oft haben Hersteller bereits einen großen Bestand an CAD- und Simulationsdaten, die für die Produktentwicklung notwendig waren und die weiter genutzt werden könnten. Für diese Fälle sind Filterprogramme notwendig, die Datenstrukturen möglichst intelligent aus plattformabhängigen Formaten in Standardformate, z.B. VRML, konvertieren.
Als weiterer Aspekt tritt die Größe der entstehenden Datei und die davon abhängige Übertragungsgeschwindigkeit in Erscheinung. Die Datenstrukturen müssen klein genug sein, um auf den verhältnismäßig langsamen Wegen des Internets übertragen werden zu können.

Für den Erfolg eines Formates ist nicht nur die Qualität der Darstellung entscheidend. Wichtig ist auch die einfache Integration von Browsern auf den verschiedenen Endsystemen und deren einfache Handhabung.
Eine weitere Anforderung resultiert aus den für viele Anwender ungewohnten Navigieren im 3D Raum mit 2D Eingabegeräten (Maus). Dies muss daher möglichst intuitiv und ergonomisch erfolgen, da keine Standards verfügbar sind.
3D-Formate lassen sich grundsätzlich in Szenengraph und Datenfluss basierte Formate einteilen. Ein Szenengraph ist eine Baumstruktur (ein azyklisch gerichteter Graph) mit einer Wurzel, Knoten und Blättern. Jeder Knoten besitzt eine Anzahl von Feldern, die Attribute des jeweiligen Knotens beinhalten. Knoten wiederum können weitere Knoten umfassen.
Zur Laufzeit wird der Graph traversiert und zur Anzeige (Rendern) gebracht. Diese Struktur ist für viele Aufgaben sehr effektiv, kann aber für sequentielles Streaming im Internet hinderlich sein, da die eigentliche Information für die geometrischen Objekte oft erst in den Blättern liegt und damit einer zeitigen Vorschau (Preview) nicht zur Verfügung steht. Es muss erst der gesamte Szenegraf traversiert werden.
Nicht Szenengraph basierte Formate sind meist Datenfluss orientiert. Dies bedeutet, dass die Objektdaten mit ihren Eigenschaften und Animationen sequentiell angeordnet werden. Man kann nach entsprechender Aufarbeitung parallele, möglicherweise komprimierte Datenströme erzeugen , die z. B. die Objekt- und Texturdaten im Wechsel nebeneinander ablegen. Zum zeitigen Preview braucht dann nicht der gesamten Datenstrom traversiert werden.
Zwischen beiden Ansätzen sind Kompromisse möglich, wie sie z. Z. vom im X3D Konsortium vorgeschlagen werden. Einen kurzen Überblick gibt die nachfolgende Tabelle.

Szenengraph	Sequentiell
VRML	Viewpoint
Java3D	QuickTime
X3D	

Tabelle 1: Klassifikation

3 VRML und X3D

3.1 Entwicklung

VRML (Virtual Reality Modelling Language) [4][22] ist eine der ältesten 3D-Beschreibungsformate für das Internet und liegt schon in der 3. Generation vor. Das von SGI entwickelte Dateiformat „Open Inventor“ [30] war Vorläufer für die erste VRML-Version. Von ihm wurden wesentliche Strukturen für VRML abgeleitet. VRML ist eine Scriptsprache und erzeugt eine ASCII Datei, die komprimiert werden kann. Mittlerweile existiert VRML’97, welches die überarbeitete Spezifikation 2.0

zum ISO-Standard machte. Im September 2000 wurde eine Spezifikation VRML 200x [22] veröffentlicht, die wiederum in den nächsten Jahren zur Grundlage einer nachfolgenden VRML-Version werden wird.

1994	VRML 1.0, Virtual Reality **Markup** Language	ausgehend von dem SGI- Inventor Format • statische 3D Anwendungen
1996	VRML 2.0 Virtual Reality **Modelling** Language	SGI- Vorschlag „Moving Worlds“ • dynamische Welten • Schnittstellen zu externen Sprachen
1997	Standard	ISO/IEC 14772-1 (VRML97)
1998	X3D Extensiv 3D	Arbeit beginnt an einem neuen Standard im WEB3D- Konsortium
2002	ISO VRML 2002	Neuer Standard unter Einbeziehung von XML, XHTML, MPEG4

Tabelle 2: Entwicklung VRML

Derzeit wird an VRML 200x/X3D entwickelt. Entscheidende Neuerung wird die Neustrukturierung des Dateiformats sein.

3.2 VRML

Zum Betrachten von VRML-Szenen wird ein normaler Internet Browser mit einem Viewer, meist in Form eines installierbaren PlugIn, benötigt.
Ein einfacher Texteditor ist ausreichend, um ein VRML-Welt als ASCII- Datei zu erstellen. Für die Be- und Erarbeitung großer Szenen ist diese Vorgehensweise nicht sinnvoll, da sie sehr zeitaufwendig und fehleranfällig ist. Zudem besteht die Schwierigkeit, den Überblick über die Welt zu bewahren. Hier helfen spezielle VRML- Editoren. Ein anderer Weg ist der Export aus CAx-Programmen, wie z.B. Maya oder 3D Studio Max. Das Ergebnis ist in der Qualität sehr unterschiedlich und bedarf in den meisten Fällen noch einer Nacharbeit.
Da jedoch die meisten CAx Anwendungen VRML unterstützen, stellt VRML den bisher einzigen weltweit akzeptierten Standard für ein 3D Beschreibungsformat dar.
Ein wesentlicher Nachteil von VRML ist die Datengröße und die damit verbundene lange Wartezeit, bis sich Szenen vollständig aufgebaut haben, bzw. traversiert wurden. Die Möglichkeit, Szenen in verschiedene Dateien zu zerlegen oder die Gzip-Komprimierung anzuwenden, bringen nur wenig Besserung.
Eine ausführliche und kritische Bewertung des Formates ist in [17] zu finden.

```
#VRML V2.0 utf8
Transform {
 children [
      Shape {
       appearance Appearance {
            material Material {diffuseColor 0 1 0 }
       }
       geometry Box {size 2 2 2 }
      }
 ]
}
```

Abbildung 2: VRML Code Beispiel

Vorteile:

- Standardisiertes 3D Grafikformat,
- viele CAD- und Design- Systeme unterstützen VRML,
- Einbindung von Programmiersprachen, z.B. Java/ Skript,
- external Authoring Interface (EAI- Java).

Nachteile:

- hohe Komplexität für den Entwickler,
- eigener, komplexer Browser ist erforderlich,
- Interaktionsmöglichkeiten auf Szenenebene sind vom Browser abhängig,
- relativ monolithische Skript Sprache,
- keine NURBS oder Splines (Vorschlag in Abstimmung),
- keine Verbindung zu WEB- Sprachen und damit wenig Integrationsmöglichkeiten,
- keine Programmierumgebung im Sinne einer Programmiersprache (API),
- nicht Streaming fähig durch die relativ freie Anordnung der Knoten,kommerzielle Schwierigkeiten im VRML- Konsortium behinderten die Weiterentwicklung.**X3D**

Das Web3D Consortium ist dabei X3D (Extensible 3D) [22][5][15] als Nachfolger des VRML-Standards zu verabschieden (Herbst 2002). Der Schwerpunkt liegt auf einer Erweiterung des Standards VRML97 zu Web3D. Ziel ist im Wesentlichen die volle Kompatibilität zu allgemein anerkannten Internet Technologien, wie XML, Java, Streaming und Kompression zu erreichen.
Bei der XML-Anwendung handelt es sich um eine Komponentenarchitektur für die Darstellung animierter 3D-Inhalte im Web. Die Größe und Komplexität der 3D-Clients ist variabel und kann jederzeit durch Plug-in-Komponenten erweitert werden. Über eine "Extension Registry", die das Web3D Consortium verwaltet, haben Hersteller die Möglichkeit, eigene Erweiterungen (Profile) beizusteuern.
Diverse Firmen – darunter blaxxun [24], OpenWorlds [26] und ParallelGraphics [23]– unterstützen die Arbeit des Web3D Consortiums.
Dessen Ziel ist es, ein Zusammenspiel von "Interoperating Standards", um webbasierte 3D-Browser plattformunabhängig durchzusetzen.

Um einen kleinen, Core genannten Kern, gruppieren sich Erweiterungen, so genannte Profile. Ein wesentliches Profile ist das VRML97 Profile, welches die Kompatibilität zu VRML herstellt. Weitere Profile sind in Arbeit.
Durch die gute Integration in Java (Xj3d) [22][24] sind bereits einige einfache Viewer/ Entwicklungsumgebungen erhältlich, die komplett in Java geschrieben wurden und frei erhältlich sind. Mittels Applet ist die Betrachtung und Manipulation von X3D Szenen möglich. Dabei kann durch Auswahl von class Files die Funktionalität dynamisch erweitert werden. Somit ist kein monolithischer großer Browser mehr erforderlich.
Die eigentlichen Szenen können dabei direkt als VRML Dateien geladen oder auch in XML- spezifiziert sein. Die XML- Spezifikation ist VRML sehr ähnlich.
In Zukunft wird es durch die enge Kooperation mit der MPEG4 Gruppe möglich sein, so genannte BIFS (MPEG-4 Binary Format for Scenes) zu nutzen, die eine drastische Reduzierung des Ressourcenbedarf durch Komprimierung versprechen. Weiterhin sollten diese Formate dann auch streaming fähig sein.

Vorteile:

- Volle Integration in XML,
- Definition eigener erweiterbarer Profile (Core, VRML, Interactive Profile) mittels DTD (Document Typ Definitions),
- Streaming und Kompression über MPEG4,
- Definition eines binären Formates mit BIFS (MPEG-4 Binary Format for Scenes),
- Spezifikation mit IDL,
- kein monolithischer Browser erforderlich,
- kein PlugIn notwendig,
- kleiner einfacher Kern (Core).

Nachteile:

- Sehr lange Entwicklung,
- Komplexität,
- bisher kaum Viewer.

4 Viewpoint VET- Technologie

Die Viewpoit VET Technologie (Viewpoint Experience Technology) [27] ist ein von Metacreations und Intel eingeführtes 3D-Streaming-Format, welches 3D-Objekte in fotorealistischer Qualität über das Internet verbreitet. Am Ende des Jahres 2000 wurde das Format in „Viewpoint“ umbenannt. Führende Hersteller, wie ADOBE, haben diese Technologie bereits übernommen.
Bei VET werden die Objekte dreidimensional und „fotorealistisch“ in die Webseite integriert und dem Nutzer schnell und in handhabbarer Form angeboten.
Interaktionen, wie z.B. das Öffnen und Schließen von Klappen sind möglich. Hilfezeiger können erklären und unter anderem die Kombination mit den richtigen Zubehörteilen ermöglichen. Texturen lassen sich dynamisch ändern. VET ist dabei

nicht für das Modellieren komplexer virtueller Welten konzipiert, sondern für deren Präsentation.
Der Vorteil von VET liegt im schnellen Übertragen der Datenstrukturen. (Streaming). Der Anwender kann schon vom ersten Moment an mit dem Objekt agieren, es drehen, bewegen, skalieren oder verändern, während es noch verfeinert und immer weiter aufgebaut wird. Die Darstellungsqualität und -geschwindigkeit passt der VET-3D-Viewer weitestgehend an die Rechner- und Übertragungsgeschwindigkeit an, um eine flüssige Wiedergabe zu ermöglichen. Besondere Serverinstallation zum Streamen benötigt VET nicht und lässt sich daher sehr flexibel auf jedem Webserver einsetzen. Allerdings sind dabei die kommerziellen Vertragsbedingungen von Viewpoint zu beachten.

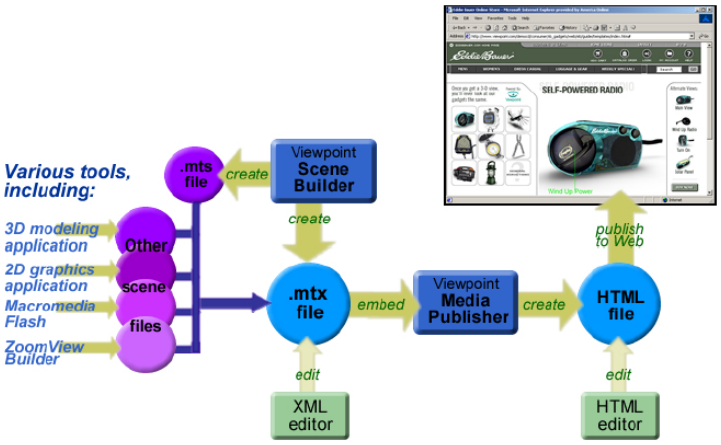


Abbildung 2: Viewpoint Struktur, Quelle: Viewpoint [27]

Zum Erstellen von MTS-3D-Daten gibt es verschiedene Möglichkeiten. Zunächst werden 3D-Modelle benötigt. Die Modelle können mittels Animationstools, wie Alias/Wavefront mit Maya oder 3D Studio Max, erstellt und über Filter exportiert werden.
Der Anforderung vieler Webapplikationen nach dynamischer Seitenerstellung wird VET durch die XML- Struktur und den gekapselten binären Dateien gerecht. Die XML- Daten können dynamisch erstellt werden. Durch unterschiedliche Binärdateien ist es möglich verschiedenartige Objekte zu erstellen. Die XML- Beschreibung ist der von X3D ähnlich. Ein MTS-3D-Szenengraph unterteilt sich in die Anzeige-, Animations- und Interaktionsgruppe. In der Anzeigegruppe können eigene Objekte erstellt werden, die nicht mehr aus der Binärdatei kommen müssen. Diese Möglichkeit ist nur ergänzend und für die Interaktion, Animation des ersten Moments gedacht. Diese Gruppe enthält außerdem die Szenenparameter und die Kameraeinstellung, die zuerst vorhanden sein müssen. Der dynamische Aspekt ist auch der Grund Interaktions- und Animationsdaten mit in die XML- truktur einzuschließen, da nur auf diese Weise große Serveranwendungen realisiert werden können.

Zu dem Kern von MTS-3D gehört ein effektiver Kompressionsmechanismus, der Texturen und Geometrien verdichtet. Dabei kommt „Trixel-NT“, ein Viewpoint-eigenes Komprimierungsformat, das auf Basis von Wavelet- Transformation arbeitet, für Texturen zum Einsatz.. Progressives JPEG als Texturkomprimierung ist optional auch möglich.
Zusammenfassend kann man VET 3D als ein ausgereiftes Format für die Darstellung von 3D-Objekten im Internet beschreiben. Die nahtlose Integration in das Webdesign mit transparenten Ebenen ist optimal für Produktpräsentationen, da sie den „Browser“ verschwinden lässt. Der Vorteil der unsichtbaren 3D-Browserintegration kann in bezug auf die Navigation nachteilig sein, da nur Standardnavigationselemente - Zoomen, Rotieren und Verschieben - vorhanden sind. Alle anderen Interaktionen müssen vom Programmieren her geplant sein.
Nachteil ist die Beschränkung auf Windowsplattformen (mit Ausnahme von MAC).

Vorteile:

- Nutzung von XML als Scenen Beschreibung,
- Streaming,
- Kompression,
- Entwicklungsumgebungen,
- ausgereifte Technik.

Nachteile:

- Kommerzielle Nutzungsbedingungen,
- kein anerkannter Standard.

5 Vergleich

Ein direkter Vergleich der Ansätze ist schwierig, da der Aufbau und die verfolgten Ziele sehr unterschiedlich sind. Untersucht werden die Formate nach Aspekten der Geschwindigkeit, der Größe, der Interaktions- und Animationsmöglichkeit. Eine getrennte Betrachtung dieser Eigenschaften ist schlecht möglich, da sie untereinander eng verknüpft sind und sich gegenseitig beeinflussen.
X3D ist noch nicht zu beurteilen, weil es zum einen noch keinen verabschiedeten Standard und zum anderen keinen Browser bzw. Erstellungswerkzeuge gibt, die über die Betaphase hinaus sind. Getestet wurde mit X3D-Edit, mit dem XML- Dateien erstellt werden können, welches aber zum Anzeigen die Daten nach VRML’97 konvertiert.
Java3D ist bei einem solchen Vergleich nicht repräsentativ, da die Datengrundlage im häufigsten Fall VRML ist und es im Internet für Produktpräsentation so gut wie keine Verbreitung hat.
Im Projektverlauf wurden Tests zwischen VRML’97 und VET durchgeführt.

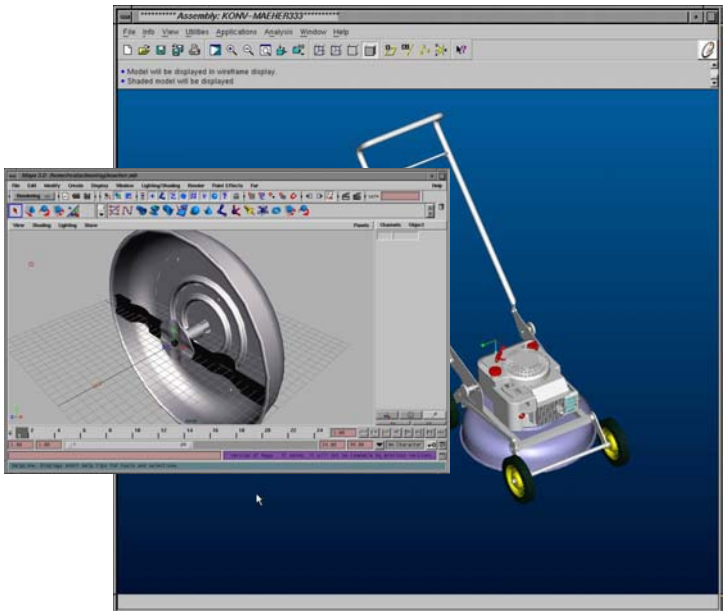


Abbildung 3: Mulchrasenmäher [29]

Es wurden die Konstruktionsdaten eines Mulchrasenmähers einer regionalen Firma [29] aus den CAD- bzw. Animationswerkzeugen sowohl nach VRML, als auch nach VET exportiert und nachbearbeitet. Animationen wurden übernommen [14].

Bewertung:
Die Dateigröße bei VET- Komprimierung lag bei 200KB, die der VRML Datei bei 2,5 MB.
VET erlaubte schon nach 10kB gelesener Daten erste Interaktionen. Nach dem vollständigen Laden war eine flüssige Interaktion neben einer modelleigenen Animation möglich. Das Einlesen der VRML Datei gestaltete sich sehr langwierig und war auf einigen weniger leistungsfähigen Plattformen kaum mehr möglich. Eine Interaktion mit dem Modell war nicht mehr sinnvoll. Die Ergebnisse waren zusätzlich von den verwendeten Browsern abhängig.
Resümierend kann man feststellen, dass VET in Bezug auf die Produktpräsentation die weit besseren Arbeitsbedingungen bietet. Es ist einfacher, kleiner und schneller. VRML hat den großen Vorteil der Skalierbarkeit für verschiedene Anwendungen. Für die normale Produktpräsentation ist dies jedoch nicht das entscheidende Kriterium. Der VRML-Nachfolger X3D scheint mit VET vergleichbar zu sein.
Allerdings muss nochmals an den unterschiedlichen Ansatz für beide Formate erinnert werden. VRML ist für die Interaktion in großen Welten entworfen worden, während VET eine engbegrenzte interaktive Produktpräsentation als Zielpunkt hatte.
Eine abschließende Empfehlung kann deshalb nicht gegeben werden, zumal die neueren Entwicklungen im X3D Umfeld, die Integration in Java und die weitverbreiteten Export- Importfilter für VRML möglicherweise eine Wiederbelebung dieses Formates bewirken können.

6 Literatur

Standards:

[1] DIN:
Grafisches Kernsystem (GKS)- Funktionale Beschreibung, DIN 66252, 1985

[2] ISO:
Graphical Kernel system for three dimension (GKS-3D)- functional description, IS8805, 1987

[3] ISO:
Graphics Programmer's Hierarchical Interactive Graphics System (PHIGS), IS 9592/9593, 1- 41988

[4] ISO:
Virtual Modelling Language, ISO/IEC 14772-1 (VRML97), 1997**Bücher:**

[5] Aaron E. Walsh, Mikael Bourges-Sevenier:
Core WEB3D; Prentice Hall, 2001

[6] Hans Lothar Hase:
Dynamische virtuelle Welten mit VRML 2.0; IX Edition, dpunkt; 1997

[7] Rolf Däßler:
Das Einsteigerseminar VRML, Taschenbuch - 416 Seiten (1999) BHV, Kaarst-Büttgen; ISBN: 3828710824

[8] W3 Consortium;
Extensible Markup Language (XML) 1.0; W3C Recommendation 10-Feb-98;

[9] Neider, Davis, Woo
OpenGL Programming Guide, 3rd Edition
(red book), Addison Wesley, ISBN 0201604582

[10] OpenGL Architecture Review Board
Reference Manual, 2nd Edition
Addison Wesley, ISBN 0-201-46140-4

Beiträge:

[11] Raßbach,H.; Pfaff,M.:
Simulation in der Produktentwicklung; Poster und Tagungsbericht zum Tag der Forschung an der FH Schmalkalden, FHS, 2000

[12] Böse,R.; Teichmüller,J.:
Digitales Produktdatenmanagement: Ein Marketing-Beispiel aus dem Mittelstand; Poster und Tagungsbericht zum Tag der Forschung an der FH Schmalkalden, FHS, 2000

[13] Böse,R.; Pfaff,M.; Raßbach,H.:
Abschlussbericht ProDat, FKZ: 1705799, FHS, 2001

[14] Christian Schmidt:
Interaktive 3D- Produktdarstellung in Internet, Diplomarbeit, FHS, 2001

[15] Volker Coors:
Des Cyberspace Kern, VRML heute: zum aktuellen Stand von X3D, iX 5/00, Seite 108

[16] Oliver Grau:
Neue Perspektiven, 3D Entwicklung mit VRML und Alternativen, iX 6/01, Seite 110

[17] Ekkehard Beier:
Lücken im Cyberspace, Virtuelle Welten: VRML-Schwachstellen und Alternativen, iX 3/00, Seite 108
[18] Andreas Rau, Frank Renner:
Durch die VR-Brille, Viewer und Autorenwerkzeuge für VRML-Szenen, iX 10/99, Seite 64
[19] Stephan Diehl:
Weltweit am Draht, Verteilte virtuelle Welten mit Java, CORBA und VRML, iX 3/99, Seite 99
[20] Tom Sperlich:
Auf neuem Kurs, Frischer Wind für 3D-Standard VRML; c't 6/99, Seite 106
[21] Dr. Jörn Loviscach:
Projekt Genesis, Lebendiges 3D mit VRML 2.0; c't 5/97, Seite 236

Internet:

[22] www.web3d.org
[23] www.parallelgraphics.com
[24] www.blaxxun.de www.vrml.org
[26] www.openworlds.com
[27] www.viewpoint.com
[28] www.opengl.org
[29] www.mws-sm.com
[30] www.sgi.com