

XML-based 3D: Content Creators, the Web, and xVRML Are Ready for Each Other

Jeffrey Sonstein

Information Technology Department, Rochester Institute of Technology, USA

jeffs@it.rit.edu

Abstract: Content creators who use the Web as a canvas may be seen as a sociological group. As knowledge of tag-based markup languages like HTML has spread through this group, content creators have become better prepared to learn other tag-based markup languages. Learning HTML did not prepare content creators for VRML, but it does prepare them for xVRML. The Web can now support delivery of high-resolution 3D to the desktops of everyday users. xVRML has significant advantages as a technology for providing virtual reality on those desktops. The specifications and documentation for xVRML have all been developed with an eye towards human readability. The structure of xVRML instance documents facilitates a constructivist approach to learning about Web 3D. The grammar and syntax of xVRML looks like any other tag-based markup language. xVRML is designed to be lightweight, simple, readable, and to encourage content creators to build on what they already know.

Introduction

Sociologists (Berger & Luckman 1966) speak of the "stock of knowledge at hand." This is composed of the things everyone in a particular social grouping "just knows." Practitioners in a particular domain may be thought of as such a sociological grouping. As knowledge of a particular nature spreads through those creating content for the World Wide Web, it becomes a part of what everyone practicing in the Web content creation domain just knows how to do. For example, as the number of people creating HTML content for the World Wide Web grows, the number of people ready to learn other things similar to HTML also grows.

When something new comes along, people often start out making sense out of it in terms of what they already know. If it is easy to relate a new thing to what is already known, then it will be easier to grasp and to internalize knowledge about and understanding of that new thing. As educational theorists say:

[...] knowledge, no matter how it is defined, is in the heads of persons, and [...] the thinking subject has no alternative but to construct what he or she knows on the basis of his or her own experience. What we make of experience constitutes the only world we consciously live in. (Von Glasersfeld 1996)

3D first entered the Web domain in a serious way with the Virtual Reality Modeling Language (VRML). When VRML 1.0 was first released in 1994, there were not many people who could write HTML and create Web pages. There was enough payoff for many content creators to learn HTML, and over the years many have. As a result of working with HTML, reading and creating content in a tag-based markup language has become part of the stock of knowledge at hand for Web-based content creators. However, there did not seem to be enough payoff for masses of content creators to learn VRML. The notation used in VRML was influenced very much by programming languages. VRML does not look at all like HTML, therefore learning HTML does not serve to form a base of knowledge upon which to learn VRML.

xVRML (Sonstein 2005) allows a content creator to express 3D virtual reality and virtual environments in a familiar-looking, tag-based markup language. xVRML is just XML. xVRML allows a content creator familiar with tag-based markup languages to easily read and understand both the node documentation and the original specification document. xVRML is defined in an XML Schema, which itself is XML and is thus readable by content creators. For example, students familiar with HTML enrolled in an introductory virtual reality class based on xVRML at RIT in the Spring of 2005 were exposed to the same materials and exercises as students who were enrolled in previous sections of the same class which were based on VRML97. Students in the xVRML-based class

went through the same materials and exercises as those enrolled in the previous VRML97-based classes, but at twice the rate of the earlier students. Apparently, learning HTML *does* serve to form a base of knowledge upon which to learn xVRML.

When VRML was first released, broadband connectivity was rare and good graphics displays were very expensive. Because 3D on the Web is both bandwidth-intensive and processor-intensive, the machines and connectivity of the average user were not up to the task. This too has changed. Broadband access is more common, and “personal computers” now do what graphics workstations used to do. As broadband connectivity and fast computers become more and more common, graphics-intensive web-based applications (like the Macromedia product line) also become more common. Some form of web-based 3D is likely to become more common as well.

The knowledge-base of content creators working in the Web environment has changed since the first attempt at popularizing 3D on the Web. The Web-and-client hardware infrastructure has also changed. I contend that xVRML is particularly well-suited to popularize 3D content on the Web because of these two changes.

Changing content creator knowledge-base

That there has been a change in who creates content for the Web is undeniable. Content creation for the Web was once seen as somewhat of an esoteric art. Content creation for the Web is now something grade school children accomplish. When you learn to read and create in a tag-based language like HTML, you are also learning perceptual sets which will allow you to learn other tag-based languages. You are not necessarily learning much which will help you to understand more traditional programming languages.

VRML reads like a programming language. Because content creators were mostly non-programmers, VRML did not look like anything most of them were used to seeing. To be fair, neither did HTML, but the payoffs for learning HTML have been immediate and were obvious to millions. The payoffs for learning something that looks like “a programming language” are not so obvious to the average person, or just as many millions would have learned to write programs as well. The seemingly idiosyncratic grammar and syntax of VRML appears to have been a barrier to wide-spread deployment of 3D on the Web. This is reported over and over by novice users posting to newsgroups and mailing lists. Programming languages are not perceived as being part of the “stock of knowledge at hand” of everyday content creators.

The notation in the following VRML code will not be hard for programmers to follow. However, an ordinary content creator who does not know how to program may well encounter some difficulties reading and understanding that VRML code. The ordinary content creator might not have such a hard time making sense of an XML tag-based markup language to do the same thing, for understanding that document type builds on what they already understand: tag-based markup languages like HTML.

This extract from a VRML97 file defines the top of a simple work-table. The code moves the table-top geometry into position, and it establishes a color for the table-top.

```
#VRML V2.0 utf8
DEF aTable Transform {
  children [
    DEF tableTop Transform {
      translation 1 0.95 -0.5
      children [
        Shape {
          appearance Appearance {
            material Material {
              diffuseColor 0.65 0.5 0.39
            }
          }
        }
      ]
      geometry Box { size 2 0.1 1 }
    }
  ]
}
```

Example 1: Extract from a simple VRML file

In contrast, the notation in the following snippet of xVRML code does not seem to be difficult for non-programmer Web-based content creators to follow. This code looks like what they are already familiar with: HTML. This code also moves the table-top geometry into position, and it also establishes a color for the table-top.

```
<?xml version="1.0" encoding="UTF-8"?>
<World>
  <children>
    <Transform name="top" translation="1.0 0.95 -0.5">
      <children>
        <Shape>
          <appearance>
            <Material>
              <diffuseColor red="0.65" green="0.5"/>
            </Material>
          </appearance>
          <geometry>
            <Box size="2.0 0.1 1.0"/>
          </geometry>
        </Shape>
      </children>
    </Transform>
  [...]
```

Example 2: Extract from a simple xVRML table

Tag-based markup language concepts are part of the stock of knowledge at hand for Web-based content creators. Learning HTML has also taught them more general lessons and perceptual sets which can be applied, consciously or unconsciously, to the task of learning xVRML or another tag-based markup language.

Changing Web Infrastructure

That there has been a change in what is considered "everyday" hardware on the desktop and bandwidth to the Web is also undeniable. From the server to the service provider to the client, the entire Web hardware, software, and firmware infrastructure has changed. When VRML was first released, my entire school (New College of California) shared one 56k line. When I run a bandwidth test at home today on my cable modem line, it says I am getting 2.4 megabits per second while downloading a file from across the continent.

My relatively "broadband" access at home today is not an anomaly, either of class or of geography. Reuters news service has been quoted by the New York Times as claiming that 85% of U.S. households "can now buy broadband services." (NY Times 2005) It is true that reasonably high-resolution 3D on the Web requires a reasonably high-bandwidth connection. It is also true that this is becoming the norm in advanced industrialized nations like the U.S.

When VRML was first released, a 300 MHz processor was the high-end norm. Today the average new consumer-grade machine runs at around 1 GHz or better. A Stanford group debated the question: "[W]ill consumer machines provide enough functionality and performance to make workstation accelerated graphics economically unviable[?]" (Owens 2005).

	uncompressed	compressed
Armchair	95,986	9,580
ChinaCabinet	134,870	8,349

ClothesCloset	165,427	8,111
Dresser	654,273	16,577
EndTable	7,713	1,426
PhoneBooth	525,187	16,465
RolltopDesk	161,503	10,041
StraightChair	35,072	4,701
TableLamp	1,973	522
Tuner	3,871	588
WorkTable	2,543	510
interior	11,439	1,395
total size	1,799,857	78,265
percent of original		4.35
percent compression		95.65

Table 1: Effects of gzip compression on xVRML file sizes

Because of changes in the Web hardware, software, and firmware infrastructure, the computer on the average user's desktop can get and display large and complex graphics files very rapidly. The average user's computer is so fast that decompression of files "on the fly" is now seen as a reasonable approach to reducing bandwidth requirements. In a recent experiment with a simple room scene, this author obtained very high compression rates (Tab. 1). The compressed version loads and displays noticeably faster than does the uncompressed version.

Creating xVRML

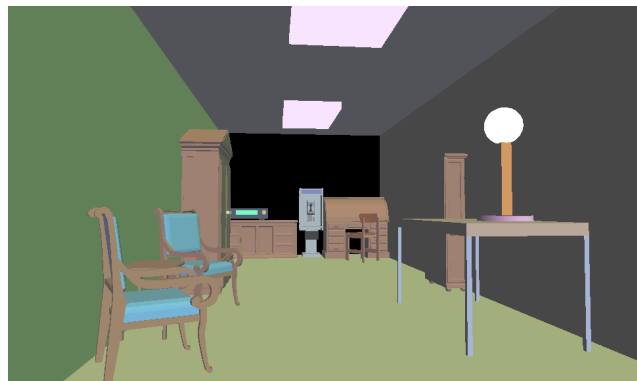


Figure 1: xVRML scene displayed in Carina viewer

The basic ideas behind tag-based markup languages have become part of the stock of knowledge at hand for content creators using the Web as a canvas. Content creators can build on that knowledge to understand a new tag-based markup language. In addition, the hardware, software, and firmware infrastructure of the Web from end to end has become powerful and fast enough that everyday users can experience relatively high resolution content. xVRML is a tag-based markup language for 3D content on the Web, but why should it be any better placed than any other such language? What particular advantages might there be to creating and promulgating Web 3D content as xVRML?

xVRML is designed "end to end" to be understandable by human beings. The base document of the xVRML specification is an XML Schema. An XML Schema is just XML, and so it is more easily read and understood by humans than a DTD. The HTML documentation for xVRML is generated from that Schema, as are example Java classes for loading/persisting an instance and other related documents. Everything revolves around the xVRML

Schema. The Schema is structured to encourage human understanding and machine readability, as well as to encourage the production of instance documents which are human readable. From the Schema to the specifications to the instance documents, all of the xVRML Project is designed to be readable and understandable by humans who are familiar with tag-based markup languages.

Teaching xVRML can be facilitated by taking a constructivist approach. The instructor can work from things the students already know: the floor and walls of a room, the geometry of a chair and a table, and, of course, a tag-based markup language. Keeping a clear separation of concerns in a scene helps the instructor divide the problem into smaller and more manageable parts.

When students are first introduced to xVRML, they may or may not know anything about 3D, much less the restrictions of the Web as a canvas. By making use of inlined files and a library of props and virtual parts, an instructor can restrict the presentation of material to just a few issues at a time. For example, learning about right-handed coordinate systems can be difficult if a good deal of other material has to be grasped at the same time. If the details of the geometry, the texture maps, the material colors, and so on, for individual objects can be hidden from the budding content creator, then it may be possible to convey the real lesson at hand (in this case, visualizing and transforming object locations in 3-space).

The following somewhat oversimplified example xVRML code is for a world consisting of some lighting, the floors and walls and ceiling of a room, and one armchair to the right rear of the room. All of the details of the lighting and the room and the armchair are hidden from the content creator at this point. The instructor and the student can focus on what is important in this early lesson: how to place an object in a scene by manipulating the translation attribute of an enclosing Transform element. The instructor and the student can use an object library and additional Transform elements to place additional objects in the room, thereby reinforcing the lesson through repetition and through constructing and manipulating an increasingly complex virtual set.

```
<World>
  <children>
    <!--lighting for the scene -->
    <Inline url="threePointLighting.xwrl" />
    <!--floor and walls and ceiling and so on -->
    <Inline url="aRoom.xwrl" />
    <!--place a chair 2 meters to the right and 5 back -->
    <Transform name="aChair" translation="2.0 0.0 -5.0">
      <children>
        <!--load the chair geometry from a file -->
        <Inline url="StraightChair.xwrl" />
      </children>
    </Transform>
  </children>
</World>
```

Example 3: A simple xVRML world

Many people learn well practicing bricolage (Turtle 1995): constructing by arranging what is already available into pleasing relationships. xVRML encourages such explorations and rearrangements of objects. All of the documentation and documents for xVRML, all the way back to the Schema itself, are human readable XML or HTML. As such, it is relatively easy for a novice or an expert to find the level of specification they need for their project. If you need to understand how the chair is made so you can make something original yourself, you can “dive down” to that level of detail. If you just need to arrange a set, you can hide the details and focus on placing the objects. At any level, you can find documentation which is readable tied to a specification which is also human readable.

Conclusion

Tag-based markup languages have entered the stock of knowledge at hand for content creators using the Web as a canvas. Content creators can build on their understanding of one such language to learn a new one for another expressive domain. xVRML allows content creators to publish Web 3D using a tag-based markup language, thus leveraging their existing skills. Content creators are better prepared to understand and to utilize xVRML than they were prepared for VRML when the Web was born.

Bandwidth availability and hardware capabilities have increased dramatically since the Web was born. Everyday desktop machines are now effectively more powerful than dedicated graphics workstations were. Content consumers are better prepared to utilize xVRML than they were prepared for VRML when the Web was born.

The xVRML specifications are understandable by humans at all levels of detail. The xVRML specifications are based on an XML Schema, so content creators can read and understand the core specifications document and so the generation of HTML documentation and other ancillary documents is easy. For the high-level arranger of objects, xVRML supports bricolage. For the low-level coder of object details, xVRML supports clearly expressed and unambiguous constraints.

Average content creator knowledge about tag-based markup languages, the infrastructure of the World Wide Web, and the xVRML Project have simultaneously developed enough to support the provision of reasonably high resolution virtual reality experiences to a mass audience on the Internet now.

References

Burger, P., & Luckman, T. (1966). *The Social Construction of Reality: A Treatise in the Sociology of Knowledge*. Garden City, NY: Doubleday.

Owens, J. (n.d.). *Untitled*. Retrieved 6 MAR 05 from <http://graphics.stanford.edu/~jowens/348c/pc-workstation.html>

Sonstein, J. (2003). *Home Page of the xVRML Project*. Retrieved 6 MAR 05 from <http://www.xvrm1.net/>

Turkle, S. (1995). *Life on the Screen: Identity in the Age of the Internet*. New York: Simon & Schuster.

Von Glasersfeld, E. (1966). *Radical Constructivism : A Way of Knowing and Learning*. Taylor & Francis.

Reuters. (2005). *WiMax May Pose Fresh Challenge to Broadband*. Retrieved 6 MAR 05 from <http://www.nytimes.com/reuters/technology/tech-telecoms-wimax.html?ex=1110258000&en=f0edbe5ad1dd9d5c&ei=5058&partner=IWON>