

Computer Graphics Teaching Support using X3D: Extensible 3D Graphics for Web Authors

Don Brutzman
Naval Postgraduate School
brutzman@nps.edu

5 October 2008

Abstract: X3D is the ISO-standard scene-graph language for interactive 3D graphics on the Web. A new course is available for teaching the fundamentals of 3D graphics using Extensible 3D (X3D). Resources include a detailed textbook, an authoring tool, hundreds of example scenes, and detailed slidesets covering each chapter. The published book is commercially available, while all other course-module resources are provided online free under open-source licenses. Numerous other commercial and open resources are available for X3D, which also serves as an interchange format. The supported course has been taught for many years, successfully introducing masters' students to the principles and techniques of 3D graphics without requiring programming experience. This course is now ready for undergraduate use. Expressing 3D graphics modeling within the domain of Extensible Markup Language (XML) for the Web is novel and has the potential to open up computer graphics to many new practitioners. This combined resource is intended broadly support computer graphics education and 3D skills for web authors.

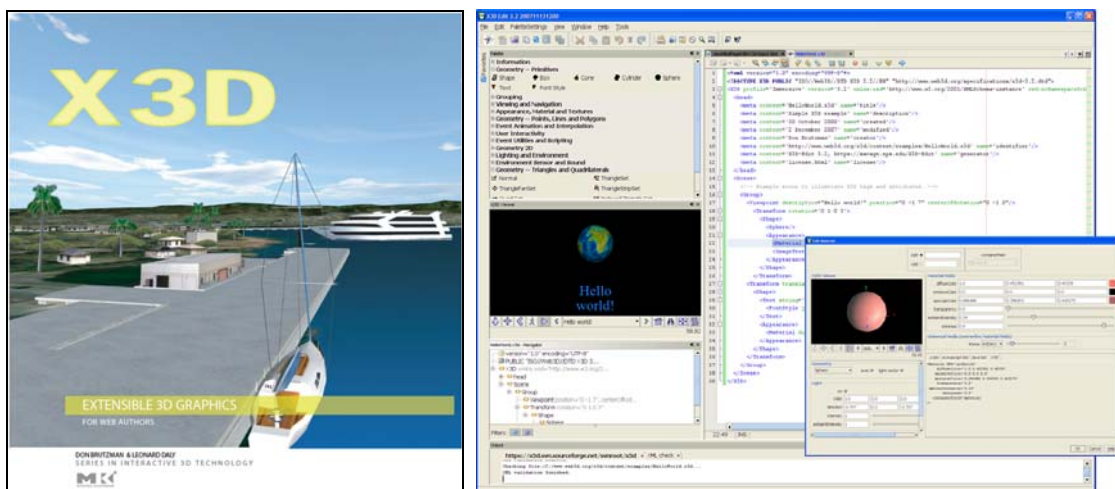
Keywords: 2D and 3D modeling, Interactive techniques, Scene graphs, Virtual Reality, Web graphics.

1 Introduction

Extensible 3D (X3D) Graphics is the international standard for Web-based graphics display and interchange. X3D is the third-generation version of the original Virtual Reality Modeling Language (VRML) which is still supported through backwards compatibility, making over 10 years of authored models still relevant and available.

This project has been accepted as a complete course module for the Computer Graphics Educational Materials Source (CGEMS) repository. Component contributions include a textbook, authoring tool, example scenes, cumulative benefits derived from many years teaching the course, and numerous supporting materials.

Figure 1 shows the cover for *X3D: Extensible 3D Graphics for Web Authors* (Brutzman, Daly 2007). Figure 2 shows an example usage of the X3D-Edit authoring tool to modify a shape's Material node.



Figures 1 and 2: Textbook *X3D: X3D Graphics for Web Authors* and supporting authoring tool, X3D-Edit.

2 Educational Goals

The primary goal of this comprehensive educational module is to show web authors how to build interactive animated X3D models. Another goal is to teach academic students the essential principles of Web-capable 3D graphics. Since X3D scene graph concepts are easily understood from the perspective of creating Extensible Markup Language (XML) documents, this material has been successfully used to teach 3D graphics to mixed classes of students who may or may not have prior programming experience. The book additionally serves as a handy reference for scene authors, explaining both the broad principles and specific details of X3D scene construction.

Web graphics has been available for over a decade, as evidenced by development of Virtual Reality Modeling Language (VRML 97), the published X3D specifications, and a dozen annual ACM SIGGRAPH Web3D Symposia. A large variety of free and commercial plugins have emerged over the years, and VRML/X3D export is commonplace from most 3D authoring tools. Nevertheless numerous “competing” commercial plugins have kept 3D graphics on the Web somewhat fractured and not nearly as commonplace as desired. The incidental role of interactive 3D graphics on the Web is underscored by the fact that most people only know of 3D via exposure to video or standalone computer games. This is further complicated by unsatisfying educational options that usually require novice graphics students to master either (a) advanced programming techniques, or else (b) a complex 3D-space authoring interface, prior to learning how to create their own 3D scenes. These are difficult prerequisites which present high barriers to learning for many students.

X3D aims to be the interchange format for the many different 3D tools and file formats that scene authors might want to use prior to publishing their work on the Web. The materials in this CGEMS contribution are aimed at general HTML/XML web authors (a large group of potential students), and not solely computer-graphics programmers or digital-arts students.

The educational goals for this project are thus quite broad. Specific objectives pertinent to the introductory X3D course follow.

- **Knowledge.** Novices learn the basic principles and practice of building 3D scene graphs. The underlying principles of 3D graphics are thus provided in a logical progression. Because multiple tools and viewers are available, students are not “locked in” to following the particular idiosyncrasies of a particular product or application. Thus they gain baseline computer-graphics knowledge applicable to further learning, and can work across the wide variety of graphics specialty areas.
- **Comprehension.** Book examples provide thorough coverage of X3D capabilities. Student comprehension is aided by having node-by-node, field-by-field tooltips provided as a context-sensitive help reference. Special editors are provided for each node to augment simple XML text editing. Tooltip translations succinctly explaining each X3D node and field are available in multiple languages. Hundreds of documented scenes encourage learn-by-example activity; students are encouraged to modify and immediately view existing scenes in order to become directly familiar with essential language constructs.
- **Application.** Student innovation and application of concepts is encouraged by urging them to build new examples that support either their theses or else other course and project work in their studies. Applying theory to practice is a recurring theme in the course. Early in the course, students are allowed to submit modifications of prior work in order to become comfortable with the creation of new scene-graph constructs. Few students feel the need to submit modifications of other authored scenes after the second or third week of teaching, instead creating new scenes of their own. Week-by-week construction of small scenes that contribute to larger mid-term and final projects is also encouraged.
- **Programming.** An interesting design aspect of X3D is that it is declarative in nature, defining animation relationships and interconnected behaviors rather than imperative step-by-step programming. Declarative design is a new concept to most students but is received well,

primarily through the identification and application of common X3D design patterns throughout. Direct imperative programming is nevertheless possible within an X3D scene by using the Script node, which can contain either ECMAScript (i.e. Javascript) or else Java. X3D also includes the notion of language extensibility by using a prototype capability, whereby an author is able to define (or retrieve from an archive) new scene-graph node definitions, which themselves are simply repeatable collections of other nodes. Scripting and prototyping are two of fourteen book chapters and the most advanced topics in the class. Within a one-quarter course, all students are able to utilize these capabilities, and perhaps half of students in class are able to master them. Further planned X3D-Edit tool support to reduce syntactic complexity may further improve programming comprehension among all students.

Hopefully most of these teaching principles and learning capabilities also apply directly to self-study students. Initial reports are positive and encouraging. Effectiveness for self-directed study has not yet been formally assessed and remains important as future work. The availability of annotated slides spanning all chapters hopefully facilitate motivated self study. The slide shown in Figure 3 is a typical example.

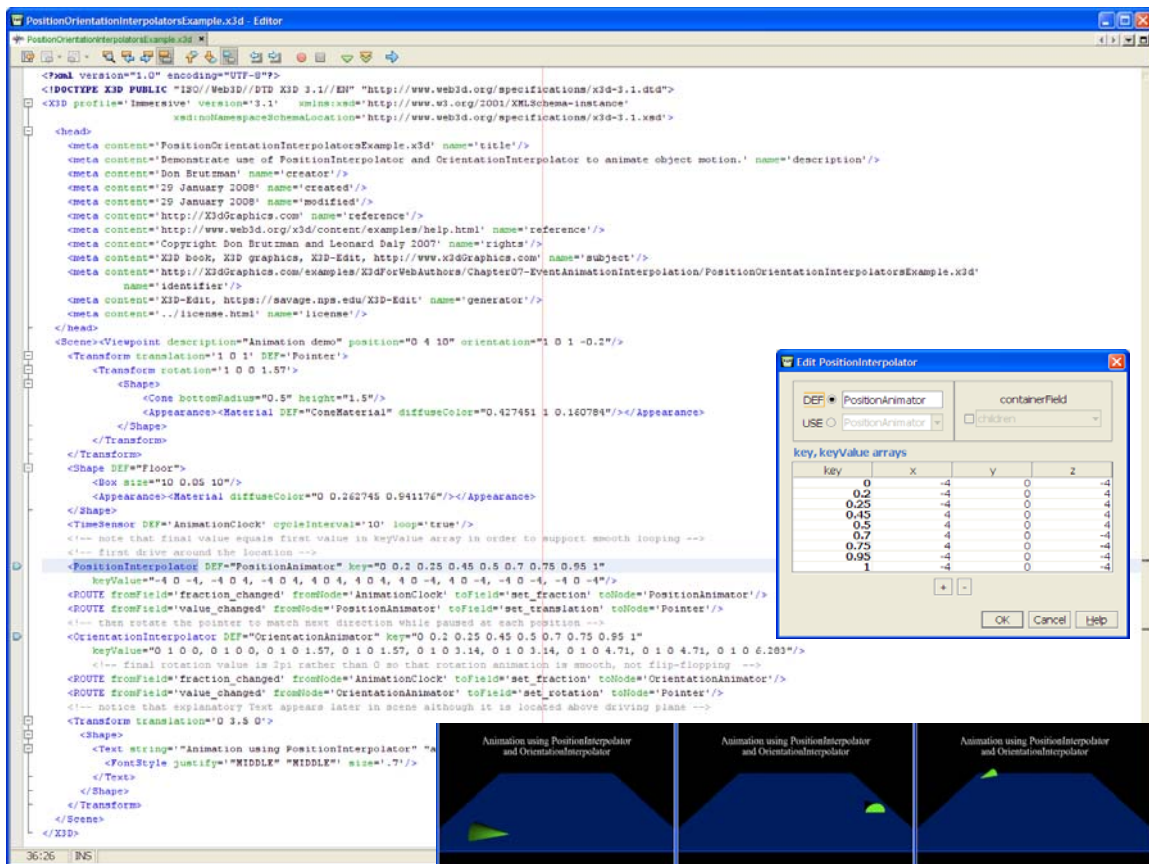


Figure 3. Representative course slide demonstrating an illustrative example scene from the book.

3 Methodology

As currently taught, students do not take tests but rather submit 8 demonstration projects during a quarter. Small scenes pertinent to the current study topics are submitted weekly to confirm that the student is keeping pace with the presentation of material. Individual midterm and final projects allow students to demonstrate proficiency. A public chat channel and an archived class mailing list each let students compare challenges and solutions as the quarter proceeds. With additional server-side setup of a

subversion system, built-in X3D-Edit support for version control facilitates students refining their projects incrementally for improved debugging, comparison and evaluation.

Course materials are usually presented with students using their laptops in class, improving their hands-on knowledge while encountering new material. A student computer lab is also a suitable venue if a projector is provided and machines are organized to support a classroom setting. Students are encouraged to experiment with their models, but remain required to include thorough document metadata (identical to HTML document metadata) in order to provide proper authoring credits and support the automatic construction of example-scene catalogs.

Students present their midterm and final projects to the class, which helps build confidence and clearly establishes “the demo” as the defining test of any 3D scene. Excellence (and occasionally, peer competition) is encouraged through the “Quantitatively Cool” metric. In the graphics community, impressed audiences and participants always say “cool!” when excellent demos are presented. This social behavior is easily measured, and it is usually painfully obvious when a demo doesn’t meet peer criteria for excellence. Hence my grading metric for any student’s classroom demo is for them to pass the quantitatively cool test. This approach is popular and effective. More classical grading approaches are certainly appropriate, as an instructor best sees fit.

Because no programming prerequisites are necessary, and based on ongoing teaching experience, the course also appears to be suitable as an undergraduate introduction to computer graphics. I am eager to work with other faculty who might pursue this possibility, and have created an archived mailing list dedicated to such efforts.

The X3D-Edit authoring tool (version 3.2) is a key part of this effort. Predecessor X3D-Edit versions 3.0 and 3.1 were the first authoring tools produced anywhere for X3D. Each tool was also open source and used in earlier versions of the class.

The current X3D-Edit tool is written in Java and based on the Netbeans Interactive Development Environment (IDE) platform (<http://www.netbeans.org>). The Netbeans plugin architecture means that X3D is now available to 300,000+ Netbeans users. More importantly, X3D-Edit also runs as a standalone application under Windows, MacOSX, Linux and (reportedly) Solaris. Features include an embedded X3D viewing window (Xj3D, <http://www.xj3d.org>), syntax highlighting, node editing wizards, tree view of the XML document, automatic updates, layout configurability, embedded help, collaborative chat, and a variety of other features. X3D-Edit remains open source with a BSD-style license. Further additions, modifications and adaptations are welcome.

Cataloged examples are automatically produced by reading the metadata descriptions contained in the X3D scenes themselves. Thus catalog addition is quite simple if students follow proper authoring conventions and best practices for documentation. All examples are available via web pages for independent viewing, either locally or online, using any free X3D plugin. Figure 4 shows an example scene.

Current work includes online video production for sessions that cover each chapter slideset and examples, providing support for X3D educators and enabling distance learning or solo study.

X3D for Web Authors book site

- Sample chapter *Technical Introduction* and unpublished chapter *Metadata*
- Course presentation slidesets
- Errata and book figures
- Mailing lists: announcements@x3dGraphics.com and educators@x3dGraphics.com
- <http://x3dGraphics.com>

X3D-Edit Authoring Tool

- XML validation using DTD and Schema, plus further validation checks during translation
- Context-sensitive JavaHelp system with integrated X3D-Edit and Netbeans editing support
- Embedded collaboration chat system (currently testing file-sharing support)
- X3D Graphics Tooltips in multiple languages: English, Chinese, French, German, Italian, Portuguese and Spanish
- X3D specifications
- <https://savage.nps.edu/X3D-Edit/X3D-Edit.html>

X3D Resources Page

- Applications, Authoring Tools, Authoring Support, Books, Conversions, Examples, License, PowerPoint, References, Training, Contact
- <http://www.web3d.org/x3d/content/examples/X3dResources.html>

X3D for Web Authors course videos

- Online video podcasts for each chapter, going through the slidesets and examples
- <http://www.movesinstitute.org/video/X3dForWebAuthors>

X3D Examples Archives (with model count)

- Basic (581), ConformanceNIST (732), VRML2.0Sourcebook (269), Savage (1189), X3D for Web Authors (238)
- <http://www.web3d.org/x3d/content/examples/X3dResources.html#Examples>

X3D Scene Authoring Hints

- Authoring, Coordinate Systems, Credits, Dates, Help, HTML, Images, Inlines/Prototypes, License, meta Tags, Naming Conventions, Scripts, URL Links, Viewpoints
- <http://www.web3d.org/x3d/content/examples/X3dSceneAuthoringHints.html>

Figure 5. X3D Graphics for Web Authors online supporting materials, many of which are also embedded in the X3D-Edit help system.

5 Conclusions

Years of work have gone into the development of the X3D language and the corresponding production of a suitable introductory course for students learning 3D graphics. These efforts are now reaching fruition. The assembled resources are now available for computer-graphics educators, hopefully streamlining introductory concepts and making 3D graphics accessible to many more students than ever before. As always, critical evaluation and steady progress remain the key to long-term success. I look forward to constructive criticism, further collaboration and broader adoption to help improve these teaching resources, expose more students to authoring their own scenes, and make 3D graphics a first-class media capability for anyone using the Web.

References

Brutzman, Don, and Daly, Leonard, *X3D: Extensible 3D Graphics for Web Authors*, Morgan Kaufmann Publishing, 2007. 468 pages, book website is <http://x3dGraphics.com>

Brutzman, Don, *Extensible 3D Graphics for Web Authors*, course slidesets, published online at <http://x3dGraphics.com/slidesets>

Brutzman, Don, X3D-Edit Authoring Tool, software distribution and user guide, online at <https://savage.nps.edu/X3D-Edit/X3D-Edit.html>

Brutzman, Don, "Graphics Internetworking: Bottlenecks and Breakthroughs," chapter four, *Digital Illusions*, Clark Dodsworth editor, Addison-Wesley, Reading Massachusetts, August 1997.
<http://web.nps.navy.mil/~brutzman/vrml/breakthroughs.html>
<http://web.nps.navy.mil/~brutzman/vrml/breakthroughs.pdf>

Brutzman, Don, Harney, James, and Blais, Curt, "X3D Fundamentals," Chapter 3, *Visualizing Information using SVG and X3D: XML-based Technologies for the XML-based Web*, Geroimenko, Vladimir and Chen, Chaomei, editors, Springer, Singapore, 2005.

Brutzman, Don, "X3D-Edit Authoring Tool for Extensible 3D (X3D) Graphics," Chapter 14, *Visualizing Information using SVG and X3D: XML-based Technologies for the XML-based Web*, Geroimenko, Vladimir and Chen, Chaomei, editors, Springer, Singapore, 2005.

Daly, Leonard and Brutzman, Don, "X3D: Extensible 3D Graphics Standard," Standards in a Nutshell column, *IEEE Signal Processing Magazine*, vol. 24 no. 6, November 2007, pp. 130-133.

Harney, James, Hudson, Alan, Blais, Curt, and Brutzman, Don, "X3D Graphics, Java and the Semantic Web," Chapter 4, *Visualizing Information using SVG and X3D: XML-based Technologies for the XML-based Web*, Geroimenko, Vladimir and Chen, Chaomei, editors, Springer, Singapore, 2005.