X3D – 3D Manmade Feature Common Data Storage Format

Katherine L. Morse, Ph.D.
Ryan Brunton
John Schloman
Johns Hopkins University Applied Physics Laboratory
11100 Johns Hopkins Road
Laurel, MD 20723-6099
858-678-0629
katherine.morse@jhuapl.edu, ryan.brunton@jhuapl.edu, john.schloman@jhuapl.edu

Keywords:
LVCAR, data, storage formats, X3D

ABSTRACT: *The Live Virtual Constructive Architecture Roadmap (LVCAR) report recommended actions to promote the sharing of tools, data, and information across the Enterprise; and to foster common formats and policy goals to promote interoperability and the use of common M&S capabilities. One of the recommended actions was to examine different data storage formats used across the various architectures to determine the feasibility of creating a set of architecture-independent common data storage formats (CDSFs). One of the categories identified requiring a common data storage format was 3D manmade features. The CDSF team performed a preliminary assessment of the ability of nine existing formats to meet 13 requirements for a manmade features format as expressed in the literature on the topic. The preliminary assessment clearly indicated that Extensible 3D Graphics (X3D) meets the requirements better than every other format except in one single case. A deeper analysis of the specific implementation of X3D was performed based upon this preliminary assessment. The results of this analysis including the specific language of the requirements are presented this paper and a recommendation is made for adopting X3D as the CDSF for 3D manmade features.*

1 Background

The CDSF Implementation Plan [1] produced by the JHU/APL team made the following general recommendations with respect to a 3D manmade features format:

1. Engage with ongoing efforts to ensure they include LVC-specific requirements:
   - Determine M&S-specific set of requirements for three dimensional (3D) manmade features representation.
   - Work with existing standards bodies to ensure requirements not yet met by their specifications are incorporated in the next iteration (Khronos Group -COLLADA1, Open Geospatial Consortium (OGC) – CityGML, Web3D - X3D).
2. Form a 3D Modeling Community of Interest (COI)
3. Identify core domain needs of M&S 3D models
4. Define specific use cases
5. Determine requirements in use cases not met by existing formats

This paper reports results from the first task including the M&S-specific requirements identified for 3D formats.

2 Existing 3D Formats

The initial Reusable Tools and CDSF Implementation Plan [2] defined the scope of this data format category as encompassing 3D models of non-geospatial data. This excludes terrain, vector map data, environmental conditions, etc., but includes 3D models that may include geospatial positioning data including buildings, vehicles, or any other model of a man-made element that may be incorporated into a synthetic natural environment at runtime that can be stored independently of its geospatial positioning. The study team identified 20 formats, 17 of which are still active:

1. X3D ISO/IEC 19775-1.2:2008
2. Virtual Reality Markup Language (VRML) ISO/IEC 14772-1:1997 and ISO/IEC 14772-2:2004
3. COLLADA
4. OpenFlight
5. SEDRIS Transmittal Format ISO/IEC 18023-(1,2,3): 2006(E)
6. CityGML OGC 08-007r1
7. Google Sketchup (SKP)
8. FilmBox (FBX)
9. 3D Studio Max (3ds Max)
10. Autodesk Maya (MA – ASCII)
11. Autodesk Maya (MB - binary)
12. Object File (OBJ)
13. AutoDesk Drawing Exchange Format (DXF)
14. OpenDWG /AutoCAD Drawing Format (DWG)
15. SolidWorks (Assembly, 2D, 3D)
16. OpenSceneGraph Binary (IVE)
17. IFC STEP ISO/IS 16739

The team collected a research corpus of contemporary work including DoD, academic, and commercial work that applied 3D formats. The purpose of this research was twofold:

- Identify community technical requirements for 3D formats.
- Determine which of the above formats are most actively in use based on citations in more than one publication.

3 Format Requirements

Review of this material led to the identification of eleven broad technical requirements illustrated in Figure 1. Additionally, the team added requirements for openness and commercial adoption since the goal of this effort is to improve adoption of common formats.

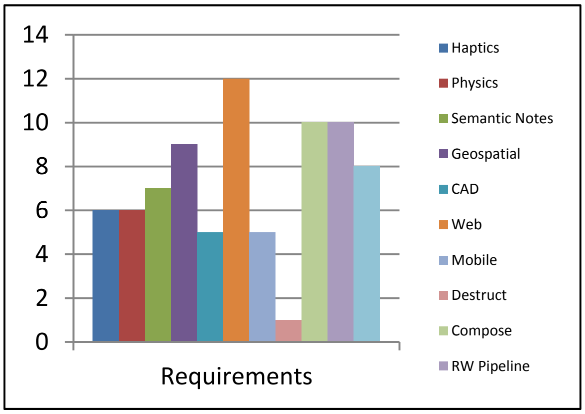

Figure 1. Technical Requirements

The JHU/APL team delivered an interim progress report [3] that included a preliminary assessment of the ability of the 17 active formats to meet these requirements. Table 1[1] summarizes the results from that report. The numbers and shading in the table indicate the degree to which each format met each requirement based upon the preliminary assessment using literature citations as the metric. Higher numbers and greener shading indicate better scores. Formats not cited in more than one publication were culled out at this stage, resulting in the nine formats assessed in the table.

4 X3D Features

Table 1 clearly indicates that X3D meets the requirements better than every other format except in one single case. A deeper analysis of the specific implementation of X3D was performed based upon this preliminary assessment. The results of this analysis

[1] Table 1 is inserted at the end of the paper for readability.

including the specific language of the requirements are presented in the following subsections. Unless otherwise noted, all evidence provided below was derived from either [4] or [5].

4.1 Haptics

Requirement: Format provides metadata associated with inputs necessary for models in this format to provide feedback to a haptic interface.
X3D Feature(s): Haptic components can be integrated into an X3D authored virtual world using an open source haptics library (H3D) via the Scene Authoring Interface (SAI). This was demonstrated within the Xj3D browser through SAI [6].

H3D extends X3D, providing the nodes and metadata necessary to support haptic-specific requirements including:

- Surfaces (smooth, frictional, magnetic, depth map)
- Interaction with five properties of haptics devices (tracker position, tracker orientation, main button, secondary button, and stylus)
- Force effects (force field, spring effect, magnetic geometry, position function, time function, and viscosity)
- Deformable shapes

See also: http://www.h3dapi.org/

H3D leverages the defacto industry standard haptic library OpenHaptics - developed and maintained by SensAble Technologies Inc. H3D is written entirely in C++ and uses OpenGL for graphics rendering and HAPI for haptics rendering [7]. However, the extensions to the X3D schema expose the full haptics API and physical properties metadata in XML, making the data suitable for storage and archiving, as well as available for use by other haptics libraries that choose to support the standard in the future.

Gaps: None

4.2 Physics

Requirement: Format provides metadata referencing the modeled object so it can be used by physics-based models (e.g., for collision response and force calculation in particle physics-based simulations).

X3D Feature(s): The X3D rigid body physics component provides the ability to influence the visual output of the scene graph in accordance to some of the laws of physics. Only the subset of the laws of physics known as rigid body physics is supported. Rigid body physics models deal with objects as solid, unchangeable sets of mass with a velocity. RigidBody nodes include:

- Angular velocity and damping
- Center of mass
- Rotation
- Inertia
- Linear velocity and damping
- Orientation and position
- Torque and forces
- Gravity

The X3DNBodyCollidableNode abstract node type represents objects that act as the interface between the rigid body physics, collision geometry proxy, and renderable objects in the scene graph hierarchy. Bounce, friction, slip, and softness are modeled.

A particle system component specifies a process for rendering such effects as fire, smoke, and snow. Although various physics models are available, they are not meant to be use for testing particle behavior models. Particle systems are designed for visual effects, not rigid analysis systems.

Gaps: None

4.3 Semantic Annotations

Requirement: Format provides a tagging metadata field for appending semantic annotations to this model for speeding discovery of appropriate models (e.g., in response to a semantically aware search algorithm).

X3D Feature(s): META statements in the header allow for unconstrained name-value string pairs. There is also X3DMetadataObject that supports nodes with multiple types of values including single and multiple floats and integers. Additionally, the MetadataSet node holds zero or more metadata nodes, allowing more complex structures than the simple parent-child relationship available with the basic metadata nodes [8].

Gaps: Although metadata is currently supported, a mechanism for referencing and embedding nodes compliant with other discovery metadata schemas would be helpful, enabling developers to tag models to be discovered by search engines using various schemas.

The authors are proposing a small addition to the X3D standard to address this gap, a new enumerated value for the meta element name attribute, metacard, whose value would be a URI for a metacard, e.g.

```
<meta name='metacard' content='https
://mscatalog.osd.mil/OSD/controller.
jsp?R=8097&hterms=pitch'/>
```

This example points to an entry in the DoD M&S Catalog that we selected at random. This approach has the disadvantages that the metacard is separated from the model and the format of the metacard is implied by the catalog that stores it. However, it has the advantage that its implementation has a smaller impact on the X3D standard which makes it more likely to be accepted. In the future, this could be expanded to specify the format of the metacard as another attribute so a tool could fetch and interpret the metacard.

4.4 Geospatial

Requirement: Format provides metadata referencing the location of a model in geospatial terms (e.g., latitude/longitude, World Geographic Survey 1986 WGS86). Models in this format can be transformed from their stored reference frame to the local coordinate system of a simulation.

X3D Feature(s): The Geospatial component includes the capability to associate real world locations with elements in the X3D scene graph via the inclusion of a GeoOrigin node. The GeoOrigin node defines an absolute geospatial location and an implicit local coordinate frame against which geometry is referenced. This node is used to translate from geographical coordinates into a local Cartesian coordinate system that can be managed by an X3D browser. The following spatial reference frames are supported:

- Geodetic spatial reference frame
- Geocentric spatial reference frame
- Universal Transverse Mercator

The following earth ellipsoids are supported:

- Airy 1830
- Modified Airy
- Australian National
- Bessel 1841 (Namibia)
- Bessel 1841 (Ethiopia Indonesia...)
- Clarke 1866
- Clarke 1880
- Everest (India 1830)
- Everest (Sabah & Sarawak)
- Everest (India 1956)
- Everest (W. Malaysia 1969)
- Everest (W. Malaysia & Singapore 1948)
- Everest (Pakistan)
- Modified Fischer 1960
- Helmert 1906
- Hough 1960
- Indonesian 1974
- International 1924
- Krassovsky 1940
- Geodetic Reference System 1980 (GRS 80)

- South American 1969
- WGS 72
- WGS 84

Gaps: None

4.5 Computer-Aided Design (CAD) Formats

Requirement: Format is recognizable by the industry standard CAD platforms and can be consumed and/or exported from them (e.g., the AutoCAD Drawing Interchange Format (DXF) (Autodesk)).

X3D Feature(s): TC184 (http://www.tc184-sc4.org) is the ISO technical committee working on visualization standardization and interoperability support for CAD models [9]. Based on their assessment of X3D, TC 184 recommended acceptance of X3D based on its fulfillment of the visualization format requirements for the Standard for Exchanged of Product Model Data (STEP), ISO 10303.

The TC184 assessment only applies to visualization (consumption) of CAD data, not production (export) of it. However, tools are available to convert from X3D to CAD formats including:

- MeshLab (http://meshlab.sourceforge.net) –DXF, STL, 3DS
- Okino Polytrans (http://www.okino.com/conv/filefrmt.htm) – DXF, STL, 3DS Max
- Ayam (http://ayam.sourceforge.net) - DXF, 3DM
- Blender (http://wiki.beyondunreal.com/Blender) – DXF, STL, 3DS

Gaps: None

4.6 Web

Requirement: Format is a lightweight data type intended for transmission and display on Hypertext Transfer Protocol (HTTP) web-based platforms (e.g., web browsers).

X3D Feature(s): X3D browsers are often implemented as plug-ins that work as an integrated part of a regular HTML web browser. X3D browsers can also be delivered as standalone or embedded applications.

Because native X3D is XML-based, and therefore verbose, the Web3D Consortium has developed an X3D Compressed Binary Encoding (CBE) [10], providing a compact transmission format that minimizes delivery size and maximizes parsing speed while following the precepts of XML.

Gaps: None

4.7 Mobile

Requirement: Format is a very lightweight data type that supports the low power/memory/display requirements of mobile devices (e.g., smart phones).

X3D Feature(s): The X3D Interactive Profile is designed for lightweight applications including mobile. This profile is a carefully chosen subset that includes seventy X3D nodes providing lightweight support for basic geometry, image textures, animation, sensing, and user interaction. Small-footprint X3D browsers show that mobile X3D applications are possible, using X3D models that work identically without modification on desktop systems [11].

The Interactive profile is more than a theoretical standard. Several vendors provide full or partial support of the profile on mobile devices.

BS Contact Mobile [12] enables the visualization of 3D Virtual Reality and Augmented Reality (AR) applications on mobile devices via X3D today. BS Contact

Mobile currently supports the Interchange profile of X3D, with partial support for the Interactive profile. The software is a stable and high performance visualization solution for handheld devices running the Windows Mobile operating system.

MobiX3D [13] is a publicly available X3D player for mobile devices. At the time the paper was written, the MobiX3D player supported a subset of the X3D Interactive profile and the full H-Anim standard. The issues encountered while developing the rendering engine were related to the limitations of OpenGL ES that implements only a subset of OpenGL functionalities.

Instant Reality has a mixed reality framework that appears to support a mobile browser, including on the iPhone [14].

Gaps: [11] identifies potential improvements for the support of mobile devices:

- A C++ version of the Scene Access Interface (SAI)
- Mobile-subset versions of SAI (EcmaScript, Java, C++)
- A floating-point version of X3D Earth model archives using carefully chosen tile sizes
- Without active dynamic server side support for clients there is little chance of getting compelling mobile content. For an X3D earth mobile app, the transformation from high precision geospatial coordinates to single precision has to be supported by the server network. It follows that the client needs to be able to request single precision content via network protocol. Specification level support for portable networked content support in X3D does not exist so this is only feasible in browser-specific implementations.

It should be noted that these identified gaps focus on optimizing runtime performance rather than extending the X3D format itself. While runtime rendering is an important factor in the adoption of mobile 3D, and X3D in particular, it is technically outside the scope of this effort that is primarily interested in data storage and interchange format capabilities. Therefore, more general-purpose solutions to the specific data challenges of mobile 3D are required that are not tied to rendering engine implementations.

Although a general-purpose solution has not been integrated into the X3D standard, significant technical progress has been made in the meantime:

- Planet 9 Studios' RayGun product [15] demonstrates floating-point precision terrain on mobile devices. Dale Tourtelotte's NPS thesis [16] describes how to auto-generate partial and full globe models from a variety of data from which it would be straightforward to auto-generate a floating-point precision globe [17].
- ISO held a workshop on mobile applications for X3D in June 2010 at which Brutzman and Behr proposed an X3D graphics profile for mobile, HTML5 and AR applications [18].
- Several companies in Web3D now agree that AR is a viable application area and believe that consolidation of diverse AR functionality is feasible. They will likely develop a mobile profile, possibly within 1-2 years [17].

4.8 Destructibility

Requirement: Format provides metadata description of the modeled object so that its decimation and destruction as the result of interaction with its external environment including other objects can be modeled. This is often a specific advanced feature of the Physics requirement above.

X3D Feature(s): Object destruction requires animation and scripting for hiding or animating the object as it's destroyed. As noted above, this is a specialty requirement, not a general capability, because there is not a consistent interpretation of how this should be represented visually because of the underlying physics of the destruction. Typically it is implemented on an object-by-object basis. However, Brutzman has developed a common authoring technique for writing these [19] as well as an implementation of an explosion triggered by a simulation's receipt of a DIS Detonate PDU [17].

In general, X3D has several options for animating destruction [17]:

- Simple scripting to receive an event and use a Switch node to hide original geometry and (optionally) show exploded version instead;
- Simple interpolation of values to receive an event and transform original geometry polygons into an exploded version instead;
- Any animation pattern can be captured as a Prototype reusable by authors in any X3D player.

Gaps: Destructibility is less about physics and more about decomposition of 3D models into constituent parts. How those parts respond interactively to the event that caused the destruction is outside the scope of a visualization format.

4.9 Composability

Requirement: Format provides metadata describing how modeled objects can be composed with other objects (e.g., composing a model of a F-16 platform with a model of a F-16 cockpit interior and a model of an AIM-120 missile).

X3D Feature(s): Various objects of the world can be collected together in the scene graph with a grouping node (Anchor, Billboard, Collision, Group, LOD [Level of Detail], Switch, Transform). Grouping nodes have a field that contains a list of children nodes. Each grouping node defines a coordinate space for its children. This coordinate space is relative to the coordinate space of the node of which the group node is a child. Such a node is called a parent node. This means that transformations accumulate down the scene graph hierarchy. An author can change nearly any part of the scene graph at run-time. The addChildren and removeChildren fields are used to ROUTE node change to a grouping node's children. Advanced scripting techniques permit the addition or removal of grouping-node children.

Through the use of DEF and USE (somewhat analogous to function calls or objects), previously described grouping nodes can be reused.

Additionally, section 4.3 describes META statements and the MetadataSet node.

Gaps: Existing X3D features provide the functional hooks for composability, but X3D does not provide a metadata specification for describing composition. This is not surprising since such a metadata specification would have to be domain-specific, e.g. describing the components of an F-16 necessary to compose a complete aircraft. This is clearly outside the scope of a standard whose expressed domain is general-purpose 3D representations.

However, work has been done to allow production of a standard for X3D metadata primarily representing platform entities that allows for model composability. The SAVAGE Modeling and Analysis Language (SMAL) [20] was developed to link specific 3D models to entities within the simulation environment. This allows the composability work done for the simulation exercise to be referenced and leveraged within X3D without burdening the 3D format directly with model semantics.

4.10 Real World Production Pipeline

Requirement: Format provides metadata intended to allow for easing manipulation of modeled objects in real or near-real-time world production pipelines of automated systems [e.g., allowing for generation of real time buildings and structures from active radar images returned from Unmanned Aerial Vehicles (UAVs)].

X3D Feature(s): This requirement should be levied on tools that extract features as part of the pipeline, generating output in a standard format. [21] describes such a process using Sketchup. There exist prototypes [22] for exporting Sketchup to X3D. There are also tools that take X3D as input to the pipeline for activities such as web publishing, e.g. WireFusion [23], including producing ontological information in web rule language (WRL) [24], e.g. Flux [25].

Gaps: None

4.11 Commercial Adoption

Requirement: Format usage is ubiquitous in commercial tools and many can already accept models in the format as is.

X3D Feature(s): Open-source as well as private commercial implementations are available for X3D including [9]:

- Xj3D, Yumetech
- BSContact, BitManagement
- Flux Player, Vivaty
- Heilan X3D Browser
- nexus3d
- instantreality Fraunhofer
- FreeWRL, Communications Research Centre – Canada
- Octaga Player, Octaga
- X3DToolKit, INRIA
- libX3D

Gaps: None

4.12 Openness

Requirement: Format is nonproprietary or a family of 3rd party tools exists for creating, reading, and exporting to this format.

Through the use of DEF and USE (somewhat analogous to function calls or objects), previously described grouping nodes can be reused.

Additionally, section 4.3 describes META statements and the MetadataSet node.

Gaps: Existing X3D features provide the functional hooks for composability, but X3D does not provide a metadata specification for describing composition. This is not surprising since such a metadata specification would have to be domain-specific, e.g. describing the components of an F-16 necessary to compose a complete aircraft. This is clearly outside the scope of a standard whose expressed domain is general-purpose 3D representations.

4.13 3D Advancements

Requirement: Format satisfies requirements that are not covered by the above requirements in this list.

X3D Feature(s): [26] describes the integration of X3D into a web-based E-learning platform.

In order to reduce spatial jitter, X3D content must be built with regionally defined GeoOrigin nodes. This approach is fine for localized regional geospatial data visualization requirements, but fails for accurately viewing data in a global context or for combining content with different GeoOrigins. [27] describes a solution to this issue.

[28] describes desired enhancements to be addressed by Web3D including:

- Real-time display and interface management

- Robust separation between APIs (immediate mode) and modules (for event publishers and subscribers)
- Reflection, for self-inspection of a scene graph during run-time operations
- Distribution across local and wide-area networks (built into the event system from the ground up)
- Dynamic scripting support
- Broad and robust testing facilities
- Authoring and annotation support within the runtime viewing engine
- Asset management, both online and offline
- Flexibility for organizations to define application-specific abstractions

The participants also listed requirements for 3D VR scenegraphs:

- Meshes, lights, materials, textures, shaders
- Integrated video, audio
- Animation
- Interaction and sensors

In addition to the findings listed above, the following ongoing X3D-related efforts were listed:

- Non-lossy round trip conversion to and from X3D and other extant 3D formats being investigated
- Research into design patterns for event propagation
- Research into a distributed, shared event model
- Sensor semantics to support real 3D input mechanisms rather than 2D devices (e.g., mouse)
- Research into view space transformations
- Research into use cases and implementations for semantic metadata

Gaps: None

5 Recommendations

Although the original intent of the JHU/APL team was to engage multiple standards bodies to incorporate M&S-specific requirements into their formats, the assessment presented in Table 1 clearly indicates that X3D already substantially meets these requirements, i.e. the distance between the current definition of X3D and a CDSF that meets the M&S community's requirements is substantially smaller than for any other format. Given this assessment, the JHU/APL team recommends direct engagement with Web3D to close the remaining gap and advocate for adoption of X3D as the M&S CDSF for 3D manmade features.

6 References

[1] Morse, K., Coolahan, J., Brunton, R., Morgan, T., Riggs, B., & Scrudder, R. (2010, June). "LVC Common Capabilities: Common Data Storage Formats Implementation Plan," Johns Hopkins University Applied Physics Laboratory, National Security Analysis Department.

[2] Morse, K., JHU/APL NSAD-R-2010-019, LVCAR Reusable Tools Implementation Plan, May 2010.

[3] Brunton, R., et al, "Live-Virtual-Constructive Architecture Roadmap Implementation, Common Capabilities – Common Data Storage Formats Progress Report", NSAD-R-2011-022, February 2011.

[4] Web3D, Extensible 3D (X3D), ISO/IEC 19555-1:2008, http://www.web3d.org/x3d.

[5] Don Brutzman and Leonard Daly, X3D: Extensible 3D Graphics for Web Authors, Morgan Kaufmann Publishers, Elsevier, 2007.

[6] Liam Kurmos, Nigel W. John, and Jonathan C. Roberts, "Integration of Haptics with Web3D using the SAI," ACM Web3D 2009, Darmstadt, Germany, June 16 - 17, 2009, pgs 25-32.

[7] SenseGraphics AB, H3D API Manual (for version 2.0), 31 October 2008, http://www.h3dapi.org/uploads/api/H3DAPI_20/docs/H3D%20API%20Manual.pdf.

[8] Don Brutzman and Leonard Daly, X3D: Extensible 3D Graphics for Web Authors, http://x3dgraphics.com/chapters/Chapter15-MetadataInformation.html

[9] Web3D, "TC184 Visualization Requirements for X3D CAD," http://www.web3d.org/membership/login/memberwiki/index.php/TC184_Visualization_Requirements.pdf

[10] Web3D, “Extensible 3D (X3D) encodings; Part 3: Compressed binary encoding,” http://web3d.org/x3d/specifications/ISO-IEC-19776-3-X3DEncodings-CompressedBinary.zip.
[11] Don Brutzman and Chris Thorne, “questions regarding X3D mobile support,” email thread, http://www.web3d.org/x3d/publiclists/x3dpublic_list_archives/0806/msg00054.html, 26 January 2008.
[12] http://www.bitmanagement.de/en/products/interactive-3d-clients/bs-contact-mobile
[13] Daniele Nadalutti, Luca Chittaro, and Fabio Buttussi, “Rendering of X3D Content on Mobile Devices with OpenGL ES,” Proceedings of the eleventh international conference on 3D web technology, Columbia, MD.
[14] http://www.instantreality.org
[15] http://www.planet9.com/raygun/index.html
[16] Tourtelotte, D.R., “X3D-Earth: Full Globe Coverage Utilizing Multiple Datasets,” Master’s Thesis, Naval Postgraduate School, September 2010, http://edocs.nps.edu/npspubs/scholarly/theses/2010/Sep/10Sep_Tourtelotte.pdf
[17] Personal conversation with Don Brutzman, April 13, 2011.
[18] Brutzman, D., Behr, J., “Proposed X3D Graphics profile for mobile, HTML5 and Augmented Reality (AR) Applications” PowerPoint presentation to ISO, June 2010.
[19] https://savage.nps.edu/Savage/Tools/Explosions/index.html
[20] https://savage.nps.edu/Savage/Tools/SMAL/SMAL.html
[21] Marc Koffi-About and Nathlaie Peypelut, “3DRV: optimal 3D Modeling Process for Urban Area Operation,” Proceedings of the 2006 Fall Simulation Interoperability Workshop.
[22] http://www.web3d.org/x3d/publiclists/x3dpublic_list_archives/0812/msg00184.html
[23] http://www.demicron.com/wirefusion
[24] http://www.w3.org/Submission/WRL
[25] http://mediamachines.wordpress.com
[26] Francesco Di Cerbo, Gabriella Dodero, and Laura Papaleo, “Integrating a Web3D Interface into an E-learning Platform,” ACM Web3D 2010, Los Angeles, California, July 24 – 25, 2010.
[27] Michael McCann, Richard Puky, Alan Hudson, Rex Melton, Don Brutzman, “Proposed Enhancements to the X3D Geospatial Component,” Web3D 2009, Darmstadt, Germany, June 16 – 17, 2009.
[28] Polys, N. F., Brutzman, D., Steed, A., and Behr, J. 2008. "Workshop report from IEEE VR 2007: future standards for immersive VR." In ACM SIGGRAPH ASIA 2008 Courses (Singapore, December 10 - 13, 2008). SIGGRAPH Asia '08. ACM, New York, NY, 1-11.

Author Biographies

Dr. KATHERINE L. MORSE is a member of the Senior Professional Staff at the Johns Hopkins University Applied Physics Laboratory. She received her B.S. in mathematics (1982), B.A. in Russian (1983), M.S. in computer science (1986) from the University of Arizona, and M.S. (1995) and Ph.D. (2000) in information & computer science from the University of California, Irvine. Dr. Morse has worked in the computer industry for over 25 years, specializing in the areas of simulation, computer security, compilers, operating systems, neural networks, speech recognition, image processing, and engineering process development. Her Ph.D. dissertation is on dynamic multicast grouping for data distribution management, a field in which she is widely recognized as a foremost expert. She is a member of Phi Beta Kappa, Dobro Slovo, ACM, and a senior member of IEEE. Dr. Morse was the 2007 winner of the IEEE Hans Karlsson Award.

RYAN BRUNTON is a member of the Senior Professional Staff at the Johns Hopkins University Applied Physics Laboratory. He received his B.S. in Computer Science from the University of California, San Diego in 2001. Mr. Brunton has extensive experience in simulation, enterprise architecture, data mining, and web technologies. He currently has a patent pending on a unique application of machine learning to the analysis of domain expert effectiveness. He is a member of Tau Beta Pi, ACM, and SISO.

JOHN SCHLOMAN is a software engineer at the Johns Hopkins University Applied

Physics Laboratory. In 2003 he completed his Master's in Computer Science at Michigan State University. In 2001, he received a degree in Systems Analysis from Miami University in Oxford, Ohio.

Table 1. Feature Requirement Support by 3D Format

	Haptics	Physics	Semantic Notes	Geo-spatial	CAD	Web	Mobile	Destruct	Compose	RW Pipeline	3D Advance
Collada (Open)	0	0	2	3	3	4	1	0	4	3	2
Extensible 3D (X3D) (Open)	2	1	2	4	4	9	2	1	5	2	5
Web3D (Open)	1	0	0	0	0	2	1	0	0	0	1
Virtual Reality Markup Language (VRML) (Open)	0	0	0	2	1	5	1	0	2	1	3
3DSM	1	1	1	1	1	0	0	0	0	1	2
Sketchup	0	0	0	1	1	1	1	0	1	2	2
Geography Markup Language (GML) (Open)	0	0	1	2	0	2	1	0	1	0	1
Shapefile	0	1	0	1	0	0	0	0	0	2	1
Unreal	1	1	0	0	0	0	0	0	0	1	0