МАТЕМАТИКА И МАТЕМАТИЧЕСКО ОБРАЗОВАНИЕ, 2004
MATHEMATICS AND EDUCATION IN MATHEMATICS, 2004
Proceedings of the Thirty Third Spring Conference of
the Union of Bulgarian Mathematicians
Borovets, April 1–4, 2004

# VIRTUAL REALITY FOR MATHEMATICAL COMPUTATIONS*

**Vladimir T. Dimitrov**

Virtual Reality Modeling Language (VRML'97) is presented. Its applicability for visualization of mathematical computations, for presentation of mathematical models, and for education and training are discussed.

**1. VRML Nodes.** The Virtual Reality Modeling Language (VRML) is an International Standard ISO/IEC 14772-1:1997 developed by VRML Consortium Incorporated [1]. It is a file format for describing interactive 3D objects and worlds. VRML is designed to be used on the Internet, intranets, and local client systems. VRML is also intended to be a universal interchange format for integrated 3D graphics and multimedia. VRML may be used in a variety of application areas such as engineering and scientific visualization, multimedia presentations, entertainment and educational titles, web pages, and shared virtual worlds. With VRML is possible to represent static and animated 3D and multimedia objects with hyperlinks to other media such as text, sounds, movies, and images.

VRML supports an extensibility model that allows new dynamic 3D objects to be defined allowing application communities to develop interoperable extensions to the base standard.

VRML built-in nodes are: Box, Cone, Cylinder, and Sphere. More sophisticated built-in nodes are ElevationGrid, Extrusion, IndexedFaceSet, IndexedLineSet, and PointSet.

The ElevationGrid is used to specify special kind of interpolated surfaces. It specifies a uniform rectangular grid of varying height in the Y=0 plane of the local coordinate system. The geometry is described by a scalar array of height values that specify the height of a surface above each point of the grid. For example, the ElevationGrid node can be used to specify such objects like mountains relief.

The Extrusion node specifies geometric shapes based on a two dimensional cross-section extruded along a three dimensional spine in the local coordinate system. The cross-section can be scaled and rotated at each spine point to produce a wide variety of shapes.

The IndexedFaceSet node represents a 3D shape formed by constructing faces (polygons) from vertices. Every face is defined by 3 non-coincident vertices. This node is used

*This work was supported by the Bulgarian National Science Fund under grant No. IS-03/2003.

to represent interpolated non-regular geometric 3D objects. The IndexedLineSet node represents a 3D geometry formed by constructing polylines from 3D vertices.

The PointSet node specifies a set of 3D points, in the local coordinate system, with associated colors at each point. Built-in nodes can be grouped to build sophisticated shapes, which can be manipulated as one.

Simplest geometrical nodes (Box, Cone, Cylinder, and Sphere) are very well supported and managed by the browsers, because there is no need of high performance computing and high resolution graphics for their visualization. The Text node specifies a two-sided, flat text string object positioned in the $Z = 0$ plane of the local coordinate system.

The other group of nodes (ElevationGrid, Extrusion, IndexedFaceSet, IndexedLineSet, and PointSet) represent shapes resulted of some interpolation on objects from real world — non regular geometric objects. For more realistic visualization of such a kind of objects - more fine interpolation must be done, which as result tends to huge VRML shapes, which are hardly manipulated by the browsers. Two kinds of computing must be done: one for generation of the node and one for its visualization. In the real word, most of objects are geometrically irregular, so to produce realistic virtual world there is a need for high performance computing for interpolation and high performance computing for visualization. In addition, there is a need for high resolution graphics for visualization of such realistic virtual worlds. Such a professional computer systems are used for production of movie pictures like The Matrix, but that are not wide spread computer systems. High performance computers needed for realistic virtual worlds will be available after 2-3 years. Let us turn to our case study — mathematical computations, mathematical models, and mathematical education and training.

Mathematical computation consists of algorithm and data. These components can be static or dynamic. Usually, algorithm (code) is static, because static code is easy supported and maintained. Such algorithms with static structure easy can be represented in virtual worlds with simple node structures like block diagrams. More sophisticated case is when the algorithm is dynamic, in such a case, there is a need for animation. Just a same case is with the data, but there is one difference: data tend to be dynamic — only in rare cases they are static.

Dynamic algorithms and dynamic data need of animation. Mathematical models tend to use simplest and regular objects, because simple models are easy studied — that is the way of scientific research. Mathematical models can be statically represented. Their visualization depends of how real world is modeled with the corresponding mathematical model. It is important simplified objects to be used in the virtual world some for representation of real ones — mathematical models give us such an opportunity. Dynamic representation of mathematical models needs of animation.

For training and education virtual worlds can be used in many areas such as training for driving license, for representations archeological objects and so on. In education on geometry at schools only simplest geometrical shapes are used. Very attractive is the visualization of mathematical problems in virtual reality. It is very important, that for this education in mathematics at schools, there is no need of high performance computers with high resolution monitors and sophisticated software. Virtual reality is available even at Bulgarian schools! Such a way of teaching mathematics is very attractive to the pupils, because they use virtual reality in computer games. Virtual reality can be used for teaching all natural sciences (physics, biology, and chemistry).

**2. VRML Animation.** The user is represented in the virtual world as avatar. It is an invisible object moved by the user. When avatar changes its position the virtual world is displayed to the user from this new position. So, moving in the virtual world user can investigate it from different positions, view points and angles. The user can jump directly to different viewpoints and shapes, or can walk or fly in the virtual world. The user even can go to different virtual worlds using the anchors put on some shapes. This is very useful feature for displaying huge virtual worlds — they can be divided into several ones and connected with anchors.

Animation of objects in the virtual world is supported with sensor nodes. They can send messages to other nodes. When a node receives some message it can change its state or visualization. There are several kinds of sensors: Collision, ProximitySensor, TimeSensor, VisibilitySensor, Anchor, CylinderSensor, PlaneSensor, SphereSensor, and TouchSensor. Usually these sensors are invisible and they are put on some visible shapes or around them. An exception is TimeSensor — it is not linked with any shape. Sensors generate their events when avatar touches them, when avatar enters in their area or when some time passes (TimeSensor).

The Collision node is a grouping node that specifies the collision detection properties for its children (and their descendants), specifies surrogate objects that replace its children during collision detection, and sends events signaling that a collision has occurred between the avatar and the Collision node's geometry or surrogate.

The ProximitySensor node generates events when the viewer enters, exits, and moves within a region in space (defined by a box). The VisibilitySensor node detects visibility changes of a rectangular box as the user navigates the world. VisibilitySensor is typically used to detect when the user can see a specific object or region in the scene in order to activate or deactivate some behavior or animation. The purpose is often to attract the attention of the user or to improve performance.

The Anchor grouping node retrieves the content of a URL when the user activates (e.g., clicks) some geometry contained within the Anchor node's children. If the URL points to a valid VRML file, that world replaces the world of which the Anchor node is a part (except when the parameter field, described below, alters this behavior). If non-VRML data is retrieved, the browser shall determine how to handle that data; typically, it will be passed to an appropriate non-VRML browser.

The CylinderSensor node maps pointer motion (e.g., a mouse or wand) into a rotation on an invisible cylinder that is aligned with the Y-axis of the local coordinate system. The CylinderSensor uses the descendent geometry of its parent node to determine whether it is liable to generate events.

The PlaneSensor node maps pointing device motion into two-dimensional translation in a plane parallel to the Z=0 plane of the local coordinate system. The PlaneSensor node uses the descendent geometry of its parent node to determine whether it is liable to generate events.

The SphereSensor node maps pointing device motion into spherical rotation about the origin of the local coordinate system. The SphereSensor node uses the descendent geometry of its parent node to determine whether it is liable to generate events.

A TouchSensor node tracks the location and state of the pointing device and detects when the user points at geometry contained by the TouchSensor node's parent group.

When a sensor generates messages these messages are send to nodes hardwired to them with ROUTE statement. There is one special kind of node Script, which can directly send messages to nodes. Script nodes are associated with programming language code which can be in-line or is accessible through URL's. VRML support Java code, but the standard is not bounded only to Java. With sensor nodes and script nodes is achieved animation in VRML. Let us turn to our case study – mathematical computations, mathematical models, and education and training.

How it is mentioned above mathematical computations have two components – algorithm and data. Dynamic algorithms can be presented with adding new components (nodes) to the scene graph to represent new calculation units. For example, if we simulate parallel processing – new process units for newly generated processes can be added in the virtual world to animate the calculation. Dynamic data can be represented with shapes representing data containers. For example, if parallel calculation is modeled, data containers can be moved through the pipes in the processing network.

Mathematical models very attractively can be animated in virtual worlds, but it is important to mention, that when the model is huge, calculation is better to be done computing server. This concept is supported in Grid computing: computations are done on computing farms, visualization is done on local desktop computers, and high throughput network connects them. If the computations are not very huge, they can be performed on the desktop computer. Mathematical models can be used in real-time or off-line. For example, in particle physics during experiments only some of events are reconstructed in real time (for experiment control purpose). All events are reconstructed off-line. In this case there is no need of high performance visualization. In some cases real-time visualization is needed, for example, when the operator must take decisions in real-time. In the last example, every particle has a number of characteristics, like color, char and many other stranger ones. These are model characteristics, but not real ones (for example, particles have no real color). Visualization of particles (with color) can improve human operator acceptance of on-going processes.

Education and training can benefits from virtual world presentations. Many mathematical problems from school course can be visualized: water pools connected with pipes; cars and truck moving from point A to point B; and so on. Very attractive is to visualize two-stroke and four-stroke engines working cycles.

Virtual worlds open new fronts for some traditional technologies for algorithm and data representation. For example, the problem with block diagrams is that it is difficult to represent in two dimensional surface complex algorithms — there are many crossing and overlapping line, objects etc. 3D virtual worlds offer new opportunities for representation — new dimension and navigation. Another example is entity-relationship diagrams used for data base modeling. When the database is real sized these diagrams are very difficult for investigation. Entities usually are simplified — they are not represented with their attributes. In virtual world is possible to put all attributes in every entity and to investigate its details. Relationships between entities can be represented without any crossings and overlappings.

**3. VRML for Data Visualization.** Virtual reality is accepted usually as real world modeled on the computer. For example, such models are of: the Solar system, some terrestrial area with its environment; game models, which are some terrestrial area with fantastic elements and etc. There are no investigations on virtual reality for more abstract models, like data models (object-relational, relational, entity-relationship, object-oriented), computational models (dataflow, finite state automata, Petri nets, CSP).

Preliminary investigations in area of data models show interesting results. Data in relational model of data is presented as tables. In virtual reality data can be presented in many different ways.

Data base of drugs contains several tables which are intended to represent all information about any drug, such information is commercial name, ATC code, drug form, international nonproprietary name, manufacturer, manufacturer country, drug components with quantities. The end-user needs to see together all information about any distinct drug. This means that in virtual reality information about any distinct drug must be shown in one shape. For example, such a shape is Text. If some editing on drug information is supported then one Text shape for the drug is not enough. Every part of above mentioned information can be bounded with sensor acting with scripts for editing purpose. How appropriate is this approach in case of real huge data bases need of more investigations. There are several other alternatives to support editing of data.

The next problem is how to situate drugs in the space. Drugs can be classified in four main ways by commercial name, by international nonproprietary name, by ATC code, and by vendor. These classifications can be represented in different virtual worlds. In every world drugs are arranged in columns in depth, where drugs in the column have the same classification characteristic. For example, in the world of commercial name classification there are 24 columns – one column for every letter; the drugs are arranged in columns by first letter of their commercial name; in the column drugs are arranged lexicographically. The first problem that arises is that when the user look at column visualization is very bad, because background of Text shapes is transparence. It is better if some non transparence Box is right behind the Text shape. Now the user can see fully the Text shape without problem.

How the user can navigate in this data? It is better to fly over the columns. The distance between the drugs and the trajectory of flight must be enough to see in depth many drugs, and when cross one drug fully to see the next. The user can jump directly to chosen drug in it is in its view. Better navigation can be achieved if starting Viewpoints are put in front every column. Another good practice is to put upper at the beginning of every column some Text shape with different color of the text. These texts must be visible from long distances. In front of all columns navigation table can be put. This table contains letters with anchors to Viewpoints in front of every column. There are some more ways to improve navigation, but they need more investigation.

This kind of navigation is supported for static worlds, but the world can be changed. For example, column can be rearranged, only part of drugs can be selected, i.e. the world can be reconstructed with relation algebra operations. This topic is under investigation.

**4. Conclusion.** Virtual reality is very attractive, but there are still limitations for its wide use. If the virtual world is build of simple shapes it is effectively managed by browsers — there is no need of high performance computers and high resolution monitors. Mathematical models use simple objects, which can be represented with simple shapes.

For training and education VRML can be used on wide spread computers now days in Bulgaria.

What is really needed to run VRML?

1. Computer that runs Windows'95;
2. Operating systems Windows'95 or letter;
3. Internet Explorer, Netscape Navigator or other browser running under above mentioned operating system;
4. Java run-time engine;
5. VRML plug-in for the browser, such as Cortona, or other one.

This configuration is wide spread now days in Bulgaria — at schools and home computers. It is possible VRML to be used under Linux or other operating system.

VRML is web-based file format, so it can be used on the web.

## REFERENCES

[1] http://www.web3d.org

Vladimir Dimitrov
POB 1829
1000 Sofia, Bulgaria
e-mail: cht@fmi.uni-sofia.bg

## ВИРТУАЛНА РЕАЛНОСТ ЗА МАТЕМАТИЧЕСКИ ИЗЧИСЛЕНИЯ

**Владимир Т. Димитров**

Представен е Virtual Reality Modeling Language (VRML'97). Разгледана е неговата приложимост за визуализация на математически изчисления, математически модели в образованието и в обучението.