ANALYSIS OF VIRTUAL DEVELOPMENT TOOLS FOR THE VIRTUAL PROTOTYPING OF A MOBILE AGRICULTURAL PLATFORM

Heleno Murilo Campeão Vale
Universidade de São Paulo (USP) – São Carlos
Heleno-usp@uol.com.br

Rafael Vieira de Sousa
Universidade de São Paulo (USP) – São Carlos
rafael@cnpdia.embrapa.br

Arthur José Vieira Porto
Universidade de São Paulo (USP) – São Carlos
Ajvporto@sc.usp.br

Abstract: *The Virtual Reality (VR) is one of the main areas of technology. Lately, the VR is going beyond its main area of application, the entertainment, being widely used in the academic and industrial areas. One area of VR that has becoming more important to engineering is the Virtual Prototyping (VP). Normally, the physical prototyping of a product can be very expensive and impracticable for some projects. In these cases, the VP can be used to prevent unnecessary expenses, reaching satisfactory results. This work presents a tool comparison and the development of a mobile agricultural virtual platform. In the initial stage some tools for virtual development were analyzed. The tools were tested in accordance with the main necessities of a vehicle's project: implementation of distributed center of mass, creation of irregular terrain, traction control, torque and other essential physical characteristics. The analyzed virtual development tools were: Webots, WorldUp, EON Studio 4.0, EON Professional and 3DCanvas. During the evaluation, it was perceived that each tool has its own characteristics and that some of them doesn't have the necessary attributes for the VP of an agricultural vehicle. The EON Professional was chosen as the more adjusted tool for the implementation of the project. After the tools' comparison, the main characteristics of the implemented virtual environment were described.*

Keywords: *Virtual Reality, Virtual Prototyping, Virtual Development Tools.*

1. Introduction

The Virtual Reality (VR) evolved from applications exclusively related with entertainment, for a tool of support in diverse areas, as medicine, architecture, engineering etc. Specifically, the activities of conception, preliminary project and tests of new products can widely be benefited by the use of VR systems (Palma, 2001), (Porto, 2000), (Kirner, 1999). Generally companies use physical prototypes, and more currently digital prototypes, for tests in new products, however frequently the cost of these prototypes is impracticable, or in the case of the digital ones it does not allow the accomplishment of the tests (Zorriassatine, 2002). With the advance of the computer science technology and the reduction of the prices of I/O devices, some companies are changing physical and digital prototypes for virtual prototypes.

As the virtual prototyping of a product it is not a simple task, the choice of the more adjusted tool to the project is extremely important. This way it is possible to save time and obtain a virtual prototype that simulates all the characteristics of a product, allowing the necessary tests.

In this work, some VR modeling tools were analyzed with intent to choose the more adjusted tool to build a Virtual Prototype (VP) of a mobile agricultural platform. In the next topic were described some requirements of a mobile platform. In the topic 3 the main characteristics of each tool were discussed and a comparison table was created. The topic 4 shows the main aspects and difficulties of the implementation phase. The conclusions and references are listed in topic 5 and 6, respectively.

2. Requirements of a mobile platform

A VP project must have its necessities well definite before its implementation to prevent waste of time and to make it possible to choose the more adjusted tool of development. In the case of this work, the VP of a mobile agricultural platform, some characteristics are essentials:

1) As a mobile platform, the implementation of torque in the wheels is necessary;
2) The simple addition of torque to the wheels is not enough to add motion to a vehicle. The torque must work together with properties of gravity and collision, to make it possible the implementation of traction with the ground. Moreover, the creation of axles between the wheels and the base of the vehicle would be necessary, so that the movement of the wheels could dislocate the vehicle;

3) To evaluate the behavior of a mobile platform with the VP, the controls, executed by keyboard or another input device, must simulate in a acceptable form the behavior of the vehicle. These also must be simple to use;
4) An agricultural platform frequently have to work on irregular terrain. Therefore, the development software must have trustworthy functionalities for implementation of gravity and collision. Movements on irregular terrain can cause bugs in the environment (Smith, 2004);
5) For the realistic simulation of a vehicle, that has to work on irregular terrain, this must have a distributed center of mass. It is not enough to add just a center of mass to the vehicle, because many times the weight of this object can be concentrated, for example, in its engine. Without the implementation of distributed centers of mass it would not be possible to make trustworthy tests on supported degrees of inclination, behavior on irregular terrain, stability etc.

3. Analysis of virtual development tools

The VP has many different tools of development. Basically, they can be divided in generic tools and dedicated tools. Programming languages as Java, JavaScript and C++ are generic tools. Although they have the advantage to be universal, its main disadvantages are: the necessity of a satisfactory knowledge in programming and the required time to the implementation. Specific tools work with the aid of the object orientation and make it possible the creation of interfaces without programming knowledge. As the different types of VR applications differ in its necessities, it is important to make a previous and detailed study of the tools' characteristics, before the implementation of a project. In this work the functionalities of five dedicated tools had been tested, for the development of a mobile agricultural platform:

- 3DCANVAS 6.5.1.0 from Amabilis Software (http://www.amabilis.com);
- WORLDUP 3 from Sense8 (http://sense8.sierraweb.net/index.html);
- WEBOTS from Cyberbotics (http://www.cyberbotics.com);
- EON STUDIO 4.0 from EON Reality Inc (http://www.eonreality.com);
- EON PROFESSIONAL from EON Reality Inc (http://www.eonreality.com).

These tools had been chosen due to its availability. The 3DCanvas was chosen to be analyzed due to the existence of a freeware version. The softwares WorldUp, WEBOTS and EON Studio 4,0 had been analyzed due to its availability in the São Carlos Engineering School laboratory. The EON Professional tool was tested during two weeks in its trial version, that supplied a complete package of its functionalities.

3.1 3D Canvas

Just the freeware version of 3D Canvas was tested due to availability reasons. This version allows just the creation of simple 3D animations, without the power to import objects from other 3D geometry modellers. Other two versions of this software are available, but both of them require a cost for its acquisition. The PLUS Version allows to import objects from AutoCad, 3DStudio among others, and also, to export objects in diverse formats. The PRO Version supplies some other funcionalities than the PLUS version, for example, to import and to export VRML and WorldToolkit objects. Just the PRO Version allows the use of scripts, therefore in the freeware version it is not possible to interact with the scene.

This software has its major advantage on its great capacity of geometric modeling. It is not necessary another software for this kind of task. In its PRO version it is possible to program interactions with Scripts (text archives that contains programming codes) in the languages VBScript or Jscript

3.2 WorldUp

Although it is not a freeware tool, the WorldUp have some characteristics that motivate its use in the virtual environment development. One of its main advantages is its easy of use and productive interface, mainly due to the organization of its class bar in a object oriented form. All the possible types of WorldUp objects are related in this class bar, in a hierarchic form, what facilitates the visualization of the content and the understanding of the scene. For the creation of a subclass or an object, the click of a button is enough (new_object or new_sub-class). Another interesting characteristic of the objects orientation in the development of a virtual system is the inheritance. The inheritance allows that a subclass inherits all the characteristics from another, simplifying, for example, the creation of different types of objects, but with similar characteristics. Another advantage of the WorldUp, is the facility to add scripts (text files that controls behavior and interactions, with Basic syntax in the case of the WorldUp) to objects of the scene. Each object of WorldUp have its own table of properties. One of the lines of this table link the text file of a script with an object. It is enough to put the address of the text file in this line to control the object with this code. The Script is read and executed at each frame played by the environment. The Basic syntax is very simple, what facilitates the implementation and understanding of Scripts in the WorldUp. For the object modeling, this software have a proper but simple modeller. So the creation of more detailed objects must be implemented in another software, specialized on 3D geometry modelling.

3.3 Webots

Webots is a software specific for 3D mobile robots simulation. It allows the simulation of different types of robots, for example, robots moved by wheels, legs or aerial robots. Its users can create complex virtual worlds and simulate its robots inside the environment. Together with the software there are a complete library of programming that allows the users to program its own robots (generally using C, C++ or Java languages). From the control programs it is possible to read values from sensors and to send commands to the engines of the robots. The result of the robot controllers can be transferred to some real robots.

This software uses ODE library (Open Dynamics Engine) (Smith, 2004) to improve the physical simulations. It can be specified a matrix for mass distribution, attrition and elasticity coefficients etc. Moreover, it has methods for collision detection, torque etc. The collision detection of solids work by the addition of BoundingObjects in their properties. The Solid node (a node in the Webots is a function that can be added to the simulation tree) has a field called BoundingObject that allows the addition of spheres, cylinders and rectangles for the creation of the collision detection geometries.

It is possible to adjust the simulation time so that the environments can executed up to 300 times faster than real robots.

A negative aspect of this software is the lack of functionalities for 3D object creation. It does not have a geometric modeller and it just imports VRML files. The Webots is adjusted for the simulation of existing mobile robots, however it is not simple to simulate a particular robot as the mobile agricultural platform of this work.

3.4 EON Studio 4.0

The EON Studio is a 3D development tool that makes possible to users, with different levels of experience, to construct virtual environments in an easy and fast form, without the necessity of programming knowledge (EON REALITY, 2005). The EON Reality has in its web site, a freeware version of EON Studio 4.0. This version has all the functionalities of the registered version, but it is valid just for two weeks.

The structure of the software is composed by three main windows: the bar of nodes (functionalities) and prototypes, the simulation tree and the bar of properties of the objects. The nodes and prototypes supplied with the software facilitate, immensely, the creation of certain types of functionalities, as models for movement, sensors, operations etc. One of these nodes, called Script, allows the addition of controls and interactions codes in the languages VBScript and Jscript. For virtual environments that need to simulate real physical effects, the software has specialized nodes prepared for implementation of collision detection, gravity, rigid body dynamics, torque etc. Moreover, it is possible to create interactions in a simple manner, without the necessity of a computer programming knowledge. It is reached by the use of the Routes Window. In this window the nodes can be linked in accordance with its variables.

Different from the WorldUp software, the EON Studio does not have any geometric modeller available with the software. Its prototypes bar supplies just simple geometries, as spheres, cylinders, rectangles etc. These geometries can be added quickly to the scene, just by the click of a button and the configuration of its properties. However, the use of this type of geometry is not viable, because they do not adapt to certain functionalities of the EON Studio. It is highly recommended to import objects from specialized 3D modeling softwares.

When tested for the development of the mobile agricultural platform, the software presented some inconsistencies in the implementation of physical properties. The gravity and collision detection of rigid bodies (objects with mass properties, speed etc.) does not function for vehicles in a acceptable manner. Simple methods for the traction creation do not exist, the wheel attrition with the ground do not cause movement. Also, do not exist prototypes for creation of axle to the wheels. As the programming languages of the EON Studio 4,0 (VBScript and Jscript) are very simple and do not supply methods to solve these problems, this software was not enough for the creation of this project.

3.5 EON Professional

The environment of creation of the EON Professional is very similar to the environment of the EON Studio 4,0, however its functionalities are more efficient. This software uses a new system of collision detection based on the physical engine of the CMLabs, that supplies a better performance, more trustworthy, than the system of the EON Studio 4.0. The new modules of visual effect and physical properties make possible a great sensation of realism and facilitate the fast environment construction. Using the Vortex technology of the CMLabs, the dynamics module makes possible a consistent simulation of rigid bodies. Rigid bodies are objects that keep physical properties, as mass and volume. They do not change its appearance during the simulation (they do not suffer deformations). The rigid bodies dynamics in EON Professional makes possible the realistic object movement, influenced by forces, axles and interaction with other rigid bodies. Two main nodes control the physical systems in EON Professional: the RigidBody and the DynamicsManager nodes. The RigidBody nodes have specific properties of each rigid body, as mass and volume, while the DynamicsManager node represents the global properties of the scene, as gravity. The normal properties of contact in EON Professional are Friction (traction) and Restituition (elasticity). As in the real world, the contact properties depend on the characteristics of the two involved bodies in the contact. In the EON Professional the characteristics of the bodies are represented in its materials and the contact properties are defined for each pair of materials. The contact properties are defined in a text file called contact definition.

The collision detection in the EON Professional is implemented in the VR standard manner, the envolvement of an object by a simpler geometry, that not overload the system. The EON Professional supplies many types of geometries capable to involve an object, for example: spheres, cubes or more complex objects than can supply more realism to the system.

3.6 Analyzed tools comparison

A comparative table, based on the necessities to the implementation of a mobile agricultural virtual platform, of five studied tools is shown below. The more adjusted softwares, with regard to each item of the table, are signed as bold.

Table 1. Tools' comparison

	3D CANVAS FREEWARE	WORLDUP	WEBOTS	EON STUDIO 4.0	EON PROFESSIONAL
IMPORT	DirectX, BioVision, Neutral Obj. Format, MD2, RAW, Wavefront, Worldtoolkit(nff)	VRML, 3DStudio, AutoCad, etc.	VRML97	VRML 2.0, 3D Studio (.3ds), ASCII/Binary (.dxf) etc	**All that showed in EON Studio 4.0 plus SolidWorks**
SCRIPTS	Doesn't exist in freeware version. VBScript or Jscript in PRO version	VBScript	**C/C++**	VBScript, JScript and Routes Window	VBScript, JScript and Routes Window
SYSTEM REQUIRE MENTS	Pentium II 300 MHz; 64 Mb RAM; Graphic card required	Pentium, 200MB, 32 MB RAM, OpenGL graphic card	Graphic card required	Pentium III 1GHz, 256MB RAM, NVIDIA GForce3	512 MB RAM high-quality graphic card
MODELING	**Few funcionalities in freeware version. Great funcionalities on PLUS and PRO versions**	Own 3D modeler with few funcionalities	Doesn't have a geometric modeler	Doesn't have a geometric modeler	Doesn't have a geometric modeler
OTHERS FUNCIO NALITIES (PHYSICAL MODULES)	Allow just modeling and animation on freeware version	Extensive funcionalities tree, but with few physical methods	Physical characteristi cs directed to specific projects	Funcionalities to gravity, mass, collision detection etc. Poor methods to center of mass and axles	**Special modules to visual effects (shadows), physical properties (axles and distributed center of mass). New CMLabs collision detection system**

The table above shows that the EON Professional is more adjusted to the construction of a mobile virtual platform, because it imports more types of 3D files and it has more physical prototypes. Therefore the implementation of this work was developed in this environment.

4. Implementation of the mobile agricultural virtual platform

The lack of modeling resources in EON Professional implied the use of another software for the geometric modeling of the environment. The software 3D Studio Max of the Autodesk (http://www.discreet.com) supplied a simple interface and all the necessary resources for the implementation of the project. More than the platform geometries, this tool were used to create all the objects of the environment, for example, irregular terrains, crop lines, rocks etc. The acquisition of objects of free sites on the Internet helped the project to speed up.

After the object modeling phase, these files were exported from 3D Studio Max with the 3DS format. In the EON Professional 3DS files can be imported in a simple and efficient way. The imported objects are located in the simulation tree, under a frame named by the path of the original 3DS file. The grouped objects (previously grouped in 3D Studio Max) are located in sub-frames of the tree. Several geometries of the platform of this project were grouped. For each wheels of the platform was created a specific group (Roda1, Roda2, Roda3, Roda4). The two laterals of the platform were also grouped (Lateral_Dir, Lateral_Esq). The object grouping is an important method because it facilitates the positioning and movement of objects in the scene, besides it simplify possible future changes.

After the importation of objects to the simulation tree, these were resized by the function Scale. Specifically in EON Professional, when the frames of objects were scaled, these lose its physical characteristics. This software limitation implies that interactive physical objects (controlled by mass, gravity, volume etc) must be imported to EON Professional in its ideal size.

The next step after the positioning and resizing of objects in the scene is the transformation of object in rigid bodies. A rigid body for the EON Professional is an object that has values of mass, suffers the influence of gravity and can be put into motion or collide with other objects. Nor all the objects need to be transformed into rigid bodies, only those that will have to interact by means of physical properties as gravity, collision etc. A rigid body demands much more machine processing that a normal object.

The figure below shows the structure of the Platform frame and its sub-frame Rodas (Wheels) in the simulation tree.

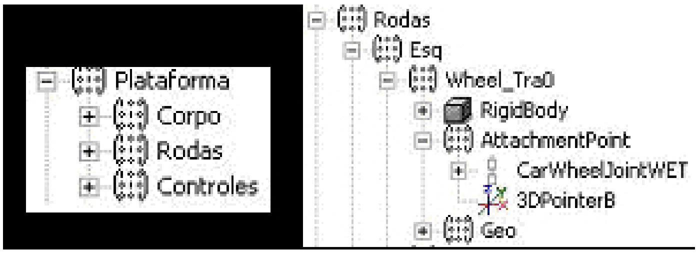

Figure 1. Platform frame structure and its sub-frame Rodas (Wheels) in the simulation tree.

In the EON Professional two nodes (functions) control the physical properties of objects: RigidBody and RigidBodyManager. The first one must be located under of frames that will contain physical properties configured by its own RigidBody node. The second one may be located under any frame of the tree. It controls the gravity and collision detection for all RigidBody nodes in the scene. The figure above shows a RigidBody node located under the left wheel frame of the platform. In this work, each one of the wheel frames of the platform received a RigidBody node with its own mass value. The Corpo (main structure) frame of the platform also received a RigidBody for the control of its physical characteristics.

More than the vehicle frames, all the terrain objects, regular or irregular, also had its properties transformed into rigid bodies. These terrain objects cannot suffer influence from gravity, so a property called Fixed had to be activated. The collision detection system in Virtual Reality demands high capacity of machine processing, due to the high level of detailing of some objects. It implies that would be impracticable to use complex geometries for the verification of collisions. The EON Professional makes it possible to choose which will be the geometry used for the verification of collisions to each rigid body in the scene (BoundingObject).

As the wheels' collision with the ground would have to move the vehicle, spheres were used as bounding objects to the wheels. Any other kind of bounding object would not function for such task. The structure of the platform used a cube as collision geometry (bounding object). However, for irregular terrains it was not possible to use simple collison geometries, because these objects have a complex structure and would lose all its characteristics. For these cases, the property "Geometry as it is" had to be used for the collision detection of irregular terrains.

Another important funcionality of EON Profession to the project was the creation of axles. There are specialized nodes for this task available in the software. In the case of this work the used node was: CarWheelJoint.

For the creation of an axle, each one of the wheels must have, under its frame, a CarWheelJoint node (figure 1). Moreover, the body frame of the vehicle must have more four of this nodes, so that pairs of linkings could be effected. The Gender property of the CarWheelJoint nodes located in the body of the vehicle must be modified for "Female". In the same way, this property of the nodes located in each wheel must be modified for "Male". Then, it is possible to create associations of the CarWheelJoint nodes with the SelectedPeer property of each node.

The JointManager node, that controls axles created during the execution of the environment, is automatically added to the simulation tree and its properties do not need to suffer any modifications. The 3Dpointer node (figure 1), when added under of a frame, facilitates the visualization of movements of the related frame, by the addition of axles X, Y and Z to its center (these axles do not influence the movement of the body, serving only for visual aid).

After the linking of all joints, the DynamicsManager node must be configured so that the platform contact with the ground works in a real manner. The properties of this node were configured as following: the gravity value was modified to "X=0, Y=0, Z=-9.81" that simulates the Earth's gravity. The elasticity coefficient was modified for "0.1" (the range is from 0 to 1). The smoothness of the collision defines how much an object can penetrate another object in a collision, its value was modified for “0.0001” (objects practically do not cross one to another). Besides all its characteristics, the DynamicsManager node also controls the levels of object traction. The property Adhesive Force was set to “0”. The FrictionType property defines if the traction will be applied in two or just in one axle (X or X, Y). In the case of a vehicle, as the agricultural platform, the traction must have the value “Friction2D”.

The FrictionModel property is used to define if the traction will take in account, or not, the mass of the object, ScaledBox or Box respectively. The second type takes less processing capacity of the system, but it is less realistic. The model used for this work was the ScaledBox. The FrictionSlip property adds landslide effect to the traction. This property was set to “0” as well as the FrictionSlide property.

The traction force is defined by the ScaledBoxFriction value (range from 0 to 1). In the created environment this property had an initial value set to “1”. However, in contact with the object “water”, this value is always modified, together with the values of Slip and Slide, to create a sensation of landslide to the platform. The landslide model will not be explained with details in this work.

No programming had been used in the project until this moment. However, for the creation of the platform movement controls, the two types of programming existing in EON Professional were used: Scripts and Routes. The input device used for the platform control was the keyboard. The Mouse controls for the movement of the camera in EON Professional (by the use of the Walk node). Under the platform sub-frame called “Control” (figure 1) a Script node was added. Also, six KeyboardSensor nodes were added to this frame. These KeyboardSensor nodes can be used in the routes window for execution of actions, activated with keyboard keys. The properties of these six nodes were configured, respectively, as: Right_Key, Left_Key, Up_Key, Down_Key, Space_Bar and Return.

In the Script node, six input data variable were created (boolean variables – True or False values) with names that make simple the understanding of the future associations with the keyboardSensor nodes. After that, the Script node and the six KeyboardSensor nodes were dragged to the routes window, so that its linkings could be effected. In this window, each one of the KeyboardSensor nodes had its On_KeyDown property linked to its respective input variable on the Script node (figure 2). In this way, each keyboard keys, previously configured, would activate a procedure inside the Script node. Each variable name is related to a procedure. The figure below show the use of the routes window:

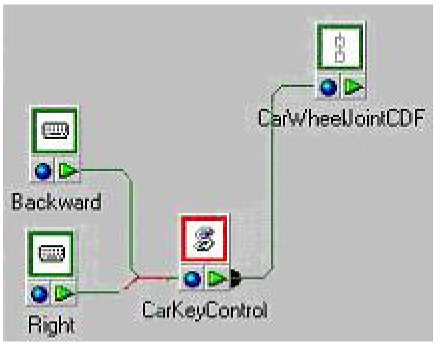

Figure 2. Routes Window example.

After the Routes creation, the Script code was written. For each one of the variables a procedure of same name was created. Each node used by the code have to be declared in the beginning of the Script. An example of declaration for the Force node is: “set ForceNode=EON.FindNode("Force")”. An example of association between an variable and the ForceVector property of the Force node is: “set Force=ForceNode.GetFieldByName ("ForceVector")”. This last one must be written below the node’s declaration. With these declarations is possible to apply a force to a body modifying the variables values. A procedure called “Initialize” is executed before any another procedure and can be used to configure the initial states of variables. The movement script of the platform modified values of Torque and Force node associated to the rear wheels, by the keyboard keys “Up” and “Down”. It would be possible, for example, to implement a 4x4 traction associating the Torque and Force nodes to all the wheels and modifying its values. The keys “Right” and “Left” were associated with the front wheels and control the platform orientation.

5. Conclusion

The data used in the project were taken from the FAPESP process: 2003/06582-0 "Autonomous Agricultural Vehicle (AAV): A platform for development of independent navigation technologies and data acquisition, in precision agriculture". Although this work used just data relative to the first draft of the AAV, some tests were executed and some conclusions, about the platform characteristics, were obtained. One of the platform characteristics tested in this work was the repositioning of its laterals to test the platform stability. Another identical virtual environment was created for this task, however the platform laterals were repositioned. The figure 3 shows the repositioned laterals.

Figure 3. Platform with repositioned laterals.

Controlling the new platform over the irregular terrains of the environment, some difficulties about its stability were evaluated. Different from the original platform, when put into motion over the slopes created in the system, the new platform frequently lost its stability. Three slopes were created in the system. The values of its angles were: 40, 50 and 60 degrees. When the original platform (without the repositioned laterals) was put into motion over these angles of inclination, the results were the following ones:

- 40 degrees: the platform never fell down;
- 50 degrees: the platform fell down depending on the speed values;
- 60 degrees: the platform fell down frequently even in low speed values.

Another test executed in the environment was related with levels of platform slide, when in contact with watered terrains. In these cases, the traction variable of the DynamicsManager node was changed from "1,0" to "0,7" and the Slide property increased from "0/0" to "0.5/0.5" (X and Y axles). With this test it was possible to predict some behaviors of the platform even without exact data to the slide values.

With this virtual environment, the platform behavior over some obstacles, that make its movements impracticable, could be tested.

Other kinds of test could have been executed with this virtual environment, such as the control of tricycles, the efficiency of the 4X4 traction etc.

During the project development, some deficiencies (bugs) of this software were found. When used for a long period, the run-time environment of EON Profession (as well as the EON Studio 4,0) became inconstant, mostly in rigid bodies collisions, forcing its reset to normalize its behavior.

Ma (2004) says in his work that one of the factors that limit Virtual Prototyping is the fact that CAD models (Computer-Aided Design) always must be created with accurate dimensions, however, these dimensions are not specified with precision in the conceptual level of the project. This factor observed by Ma (2004) was also verified in this work, because the used data were taken from an first draft of a project. Despite this lack of data, some tests executed in the virtual environment helped to foresee the real behavior of the mobile agricultural platform, without the necessity of physical implementation.

Although the good EON Professional results about VP, two factors can make impracticable its use. The main one is the high cost of this kind of specialized tools. The other factor could be verified during the development of the project. Some operations, as shade application or the considerable increase of the object number in the scene, speed down the environment even in a high capacity computer: XEON 3.06 GHz, 3,5 GB RAM, NVIDIA FX 1000 card. Therefore, to use all the funcionalities of this kind of software it would be necessary a special computer, what could make impracticable a project, due to its high cost.

6. References

3DCANVAS Documentation. Amabilis Software. Available at: <http://www.3d-canvas.info/3DCanvasDocumentation/tableofcontents.htm>. Nov. 2004.

EON Reality, EON Studio, Available at: www.eonreality.com. Jan. 2005.

Kirner, C., Introdução a Realidade Virtual, 1999. Available at: <http://www.realidadevirtual.com.br>. Jan. 2003.

Ma, W., et al., A hierarchically structured and constraint-based data model for intuitive and precise solid modeling in a virtual reality environment. Computer-Aided Design 36, 903-928, 2004.

Michel, O., Webots: Professional Mobile Robot Simulation, Journal of Advanced Robotics Systems, 2004, vol. 1, pags. 39—42. Available at: <http://www.ars-journal.com/ars/SubscriberArea/Volume1/39-42.pdf>. Jan. 2005.

Palma, J.G., Metamodelo para a modelagem e simulação de sistemas a eventos discretos, baseado em redes de petri e realidade virtual. Uma aplicação em sistemas de manufatura. 2001. 133 f. Tese (Doutorado em Engenharia Mecânica). Escola de Engenharia de São Carlos, Universidade de São Paulo – USP.

Porto, A.J.V., Palma, J.G., Manufatura Virtual, Revista Fábrica do Futuro, cap. 10, 2000.

Smith, R., “Open Dynamics Engine v0.5 User Guide”, 2004.

Sousa, R.V., et al., Robô agrícola Móvel (RAM): Uma revisão das pesquisas recentes sobre sistemas de navegação autônoma de robôs e veículos agrícolas, IV Congresso Brasileiro da Sociedade Brasileira de Informática Aplicada à Agropecuária e à Agroindústria. Anais. 2003.

Sousa, R.V., et al., Uma plataforma autônoma para aquisição de variabilidade espacial, Congresso Brasileiro de Agricultura de Precisão – Piracicaba, SP – ESALQ/USP, 2004.

WORLDUP Programmer’s Guide. Sense8. Available at: <http://sense8.sierraweb.net/products/tech_pass.htm>. Oct. 2004.

WORLDUP User’s Guide. Sense8. Available at: <http://sense8.sierraweb.net/products/tech_pass.htm>. Oct. 2004.

Zorriassatine, F., et al., A Survey of Virtual Prototyping Techniques for mechanical product development. Proc. Insto. Mech Engrs Vol. 217 Part. B: J. Engineering Manufacture, 2002.