



# **Swimming bath in VRML**

## **Documentation for semestral work in Y36MVR**

Academic year 2009/2010, WS

## Content

I. Advertisement.....	3
II. Technical description.....	4
1. Toi Toi toilet box.....	4
2. Toboggan.....	4
3. Table with parasol.....	4
III. A little help for future students.....	6
IV. Conclusion.....	7

## **I. Advertisement**

## II. Technical description

### 1. Toi Toi toilet box

Toi Toi toilet box is realized as a set of prototypes. All components were modelled directly in VRMLPad. All nodes have standard geometry, except the roof, the knob and spaces for textures, these have geometry of `IndexedFaceSet`.

Main dynamic component is the door, which carries logic of the crossbar animation. Two manipulators exist for user interaction: With `RotationInterpolator` node, whole door can be opened and closed and similar functionality has the toilet seat. The crossbar animation was mentioned before. The opening and closing of the crossbar can be invoked by pressing the button below the crossbar. Again, the `RotationInterpolator` node is used, the color of the button is changed using the `ColorInterpolator`. The animation is enhanced with different sounds for both opening and closing. These sounds are wav files with 16 bit-depth and 44 kHz sampling frequency.

Because the specification of the task contains the "Machine is working" animation, I decided to create the roof animation. It slides up and down, using the `PositionInterpolator`. Model contains main switch as well. Main switch can pause and resume all animations including the "Machine is working" animation.

Model contains two textures: The door texture, which can be changed with prototype parameter, and texture for the hole. The hole texture is a gif file, the door texture is a png file.

### 2. Toboggan

### 3. Table with parasol

Table with parasol is realized as a set of prototypes which are created entirely of standard geometry except a parasol cover and chair foots. These are made as `Extrusion` node but any of them weren't modelled in external modeller.

There are two dynamic components – parasol and chair. Manipulators are placed in the model of chair. They make the chair able to be fold because each of them matches each integrated part of the chair.

But "working-machine" animation is represented by opening and closing of the parasol. It is realized by `PositionInterpolator` and it is possible to suspend or resume it through a main switch lying on the table desk.

Behaviour of this button is same as behaviour of real button and when it is pressed, its colour will be changed in dependence on the "machine" state. Whole animation is accompanied by sound.

The model uses two textures. The first of them is a texture of wood and it is variable through the parameter called `chairTexture`. The texture called `billboardTexture` is used to cover billboard which are replacing the model when the avatar is in distance between 70 and 120 meters by it.

### **III. A little help for future students**

During the work, student has to deal with four subtasks: Modelling, prototyping, creating dynamic actions and scripting. Probably the hardest part of these four is the one not mentioned, the idea. If a student wants to ease his work a bit, he will think up a model, that will not be very hard for modelling (unless he is an experienced 3D modeller), and that can satisfy requirements of other parts of the task, mainly if there is enough parts suitable for animation or manipulation.

In the first phase, modelling, it is very useful to prepare model for easy prototyping of all parts. As soon as we have individual prototypes, the second subtask is an easy one. In this moment we might challenge the first problem, and that is different behaviour of urls, for example textures. The initial directory is selected differently when using a model or using a model as a prototype. All paths can therefore differ quite a lot.

The third subtask is connected to prototypes as well. If you had defined them using `DEF` before, routing of actions will be very easy and the development will be very fast. Some other challenges can occur when creating animations. For example, the pivot of rotation sometimes has to be moved a bit, so the pivot is placed properly.

Last subtask is programming scripts in javascript programming language. In this part, it is very important to keep look at semi-colons. One is missing and whole script is not working.

## **IV. Conclusion**

During the task we learned how to work with VRML. It showed up, that it is definitely not dead technology and for rapid dynamic scenes development is very comfortable tool. Also it is great preparation for graphics programming in OpenGL.