Czech Technical University in Prague
Faculty of Electrical Engineering
Software technologies and management

Swimming bath in VRML

Documentation for semestral work in Y36MVR

Academic year 2009/2010, WS

Jiří Chadima
chadijir@fel.cvut.cz

Daniel Kavan
kavanda1@fel.cvut.cz

Jiří Mašek
masekji4@fel.cvut.cz

Content

I. Advertisement

(following text is only the advertisement content, the actual full-featured advertisement poster can be can in "plovarna-poster.en.pdf")

You know how it is.. You need a professional looking VRML aqua park parts that wouldn't just represent your intention well, but you also want it to be sensually structured. Not to mention pretty animations too. Well, look no further, because we have a solution for you.

1. The comfort of furniture

Are you used to the comfort of the real world? Then there's no need to compromise in virtual reality. Our models will provide sufficient number of seats and a table to place objects on. Last but not least there's a folding parasol which will shield you from harmful sunrays.

As in case of all our models, this set is equipped with custom light source activated by avatar's proximity. And the best thing is: Tidying up is just as easy as use. Chairs can be folded and leaned against the table. It like one click away!

2. A safe-house

Let's not beat around the bush. Even in a virtual reality there has to be place to "rest". That's why we offer our first-class state-of-the-art ToiToi Restrooms. Thanks to interchangeable door textures, avatars can the right restroom just for her or him.

Safety is guaranteed: Every Restroom has a latch built-in. Don't worry that it squeaks and looks kind of rustic, it locks properly. And if you happen to feel alone and isolated from the outside world, there's no need to panic. Turn your look upwards and you will see either an azure sky with silver clouding or shinning stars of a summer night.

3. A water tower

What kind of a water park would it be without a proper attraction? Of course we've got one - twenty-eight-feet-tall monumental water slide that everybody loves. A color-changing elevator, a free shaped roof and a flag fluttering in the wind (or along our visitor's wishes) – it's simply perfect

You can easily tell if the toboggan is running – there's a stylish candy-caned circle decoration for that. No matter whether you're going to stop or resume it, the detonation box does the job. As a cherry on top comes the possibility of customizing the front texture of the tower.

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

The toboggan model comprises of 3 levels of detail (not counting the last empty node). They are used depending on avatar's distance.

If closer than 131 feet avatar is presented with the most detailed model version. There's a slide created using the Extrusion node (a U-shaped contour traced onto a 9-feet-long-radius spiral spine). The slide color can be customized from the outside using the `barvatela` parameter. In the middle of the slide there's a column (`Box` node) acting as a support structure linked by ropes (`IndexedLineSet` nodes). On the top a flag resides that can be freely rotated thanks to its parent `CylinderSensor`.

Another part is the tower equipped with an elevator. The walls have been created with a non-primitive `IndexedFaceSet` in order for every wall to be possible to have a different color or a texture (e.g. the front texture). The top of the tower is roof-covered. The roof's height can be changed by

dragging it up and down within a given range – this feature is powered by a `PlaneSensor`. The elevator changes its color and emits a sound while moving.

The main controller – found just in front of the entrance – is of course the (in)famous detonation box. It is `Script` driven and stops or resumes the "machine operation". When used and all animations are stopped; the best representation of that would have to be the candy-caned rotating circle on the left side of the tower.

While found in the distance between 132 and 230 feet, the model details get highly decimated: The slide turn angular, the tower is replaced by a block, and none of the previous functionality is preserved. Thereafter until the threshold of 393 feet is reached, all the visitor is given is a block as a tower and a 2D shape on a `Billboard` resembling the slide. After that there's nothing more than an empty `Group` node for the avatar to see.

No part of this model is an export from any external tool – everything has been written in VRMLPad (except for images).

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 for 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.