

## Appendix 4

### MAKING HTML PAGES AND INPUT FILES FOR THE 3D GALLERY

By Janne Pietarila

'Script 1' is a Perl script that is run twice: once for making Finnish files, and once for making English files. It takes as an input a language (Finnish or English), URL to the 'Script 2', and names of the input and output files. It works like this:

1. Print HTML header to the output HTML file (gallery\_finnish.html or gallery\_english.html).
2. Set variable ID to 1.
3. For each table row in input HTML input files:
  - A. Extract all the given information about the picture:  
name, archive no., dating, URL to the HTML file containing the picture, etc.
  - B. Transform that URL to a path in the local file system and use that path to load the picture HTML file into the memory.
  - C. From that HTML file extract URL to the actual picture file and transform that URL to a file path in the local file system.
  - D. Output to the output text file (fi\_input.txt or en\_input.txt) the following information:
    - the ID variable
    - path to the picture file
    - URL to the picture file
    - URL to the HTML file containing the picture(Each line of out text file has these field separated by tabs.)
  - E. Print HTML to the output HTML file (gallery\_finnish.html or gallery\_english.html). Example of HTML input for one picture:

```
<tr>
<td><input type=checkbox name=painting value=42></td>
      <td>Jaw of horse</td>
      <td>TYA 642: 2858</td>
      <td>980&#150;1220 jKr.</td>
      <td>Mulli site</td>
      <td>slide</td>
      <td>Antti Huittinen</td>
</tr>
```

- F. Add one to the variable ID.

4. Print HTML footer to the output HTML file (gallery\_finnish.html or gallery\_english.html).

The important thing is that 'Script 1' is run only when HTML input files have been changed.

### Creating VRML nodes from HTML files

The 'Script 2' is Perl script that handles the user input from the gallery page (gallery\_finnish.html or gallery\_english.html). The script receives from the HTML form the following information:

- the IDs of selected pictures
- the name of the input file (fi\_input.txt or en\_input.txt)
- the language (English or Finnish)
- which submit button was pressed ('Enter gallery' or 'Save configuration as HTML page')

1. Check the number of the selected pictures: if it is less than ten or more than sixteen, print the HTML page which advises the user to select suitable number of pictures.

2. Check which submit button was pressed, if it was 'Save configuration as HTML page' output an HTML page containing all the received information in the hidden input tags and instructions what to with this page. So basically the generated HTML page contains the IDs of selected pictures in the hidden tags and one submit button to enter the gallery. Exit the script.

If the script gets this far, it is known that the input is valid and the aim is to output VRML.

3. Output some static VRML (from normal VRML files) which are always the same independent input (like the island, the trees, and the sea) and some VRML PROTOs which are used later.

4. Read the input text file (fi\_input.txt or en\_input.txt) and for each ID extract corresponding picture path, picture URL, and HTML page URL.

5. For each picture:

A. A command 'identify' (part of ImageMagick, see <http://www.imagemagick.org/> for more information) is used with

the picture path as an argument to get width and height of the picture.

B. From dimensions of the picture calculate the dimensions of the border.

C. Output a Transform node with an instantiated a PROTO called 'WallWithPainting' as its child. The fields of 'WallWithPainting' include URL to the picture, URL to picture HTML page, and numbers describing the dimensions of the picture and the border. The node creates a 3 meters wide and 4 meters high wall section with a painting.

At this point the script has output a VRML world with some walls, paintings on walls, and trees on an island.

6. Output outer walls, glass walls, floor, ceiling, doors. The only thing worth noticing is that some elements have to be made bigger (an outer wall section, the glass wall, the floor, and the ceiling) and the door has to be moved depending on the number of the selected pictures. The floor and outer wall textures have to be scaled so that they always look right independent of the size the polygon they are mapped on.