1. The outcome

That's right, we begin with the ending.

CHILI Publisher holds a concept called "folding settings". These settings enable users to render a document in 3D, so the end user can see how his/her design folds.

Check the video to see how this unfolds (pun intended): https://www.youtube.com/watch?v=j2vSfYykLHY

The folding configuration is stored in an XML file.

The assets folder of this challenge contains the following file: Beerpack.xml.

As soon as this file is loaded in our software, users get an interactive 3D view similar to this:

We don't expect our users to have any 3D expertise, we just want them to reap the benefits of ours.

All they have to do is define the rectangular "panels" on top of their graphical document. For each panel they then have to determine the minimum and maximum rotation angles.

The 3D engine in CHILI publisher takes it from there and launches the visual.

Nifty, right?

2. The coding challenge

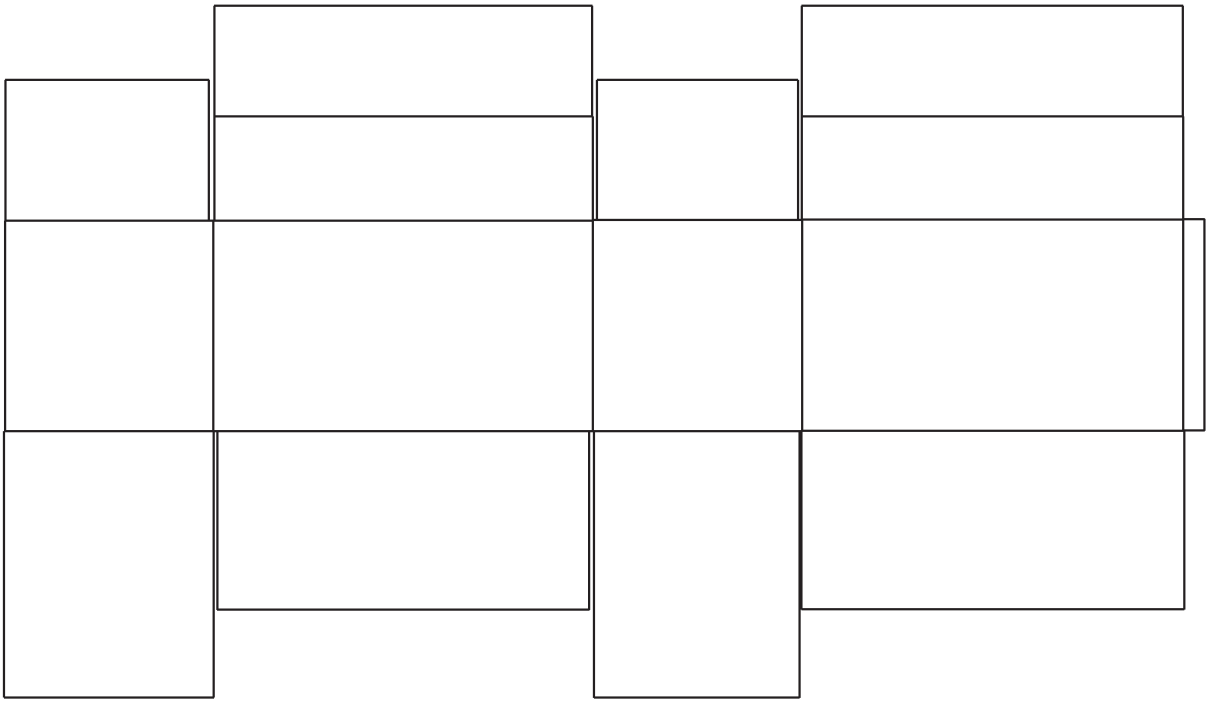
Nifty requires creative development.

What we would like you to do?

Develop a C# application that:

- Parses the supplied XML file (Beerpack.xml) and loads it into an object model
- Renders a 2D representation of the folding scheme.

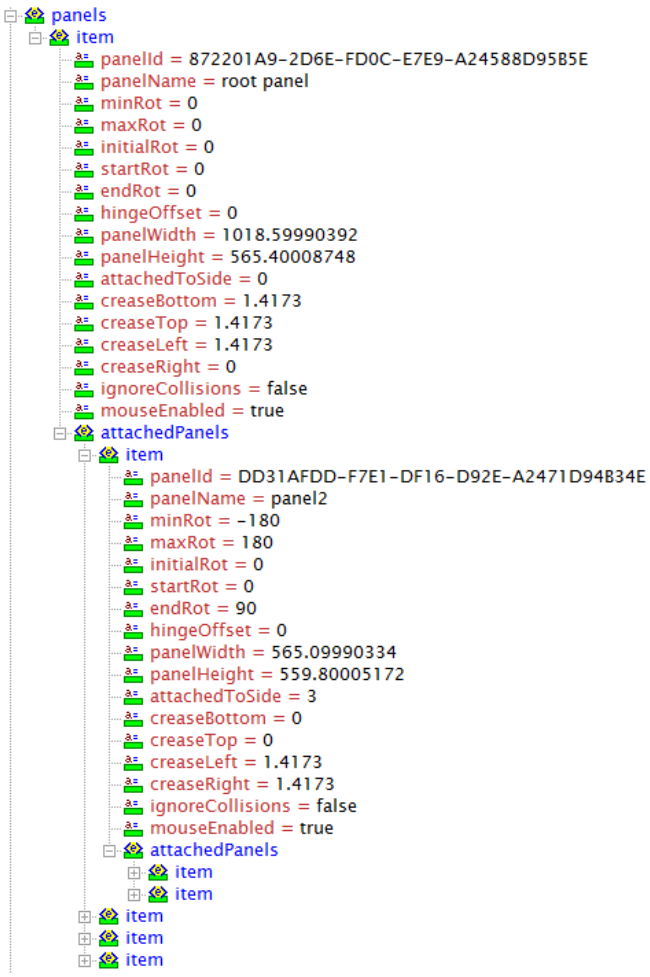
This is what the end result should look like:



3. Tech talk

You'll need these background essentials before hitting the keyboard.

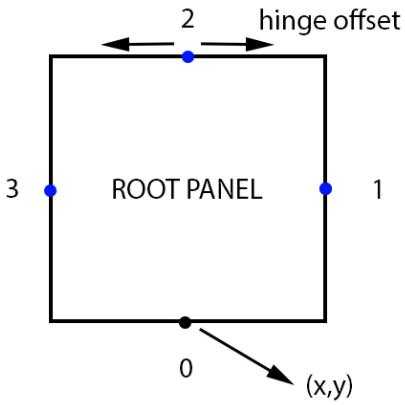
When you have opened the supplied xml file, you will notice the panels are stored in a tree structure. There is always one root panel, and each panel can have child panels. The children of each panel can be found in the "attachedPanels" child node of each panel.



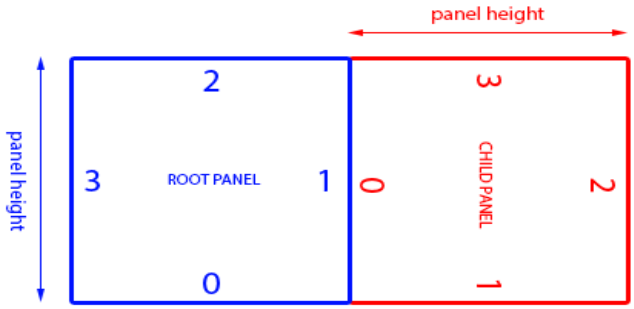
The position of the root panel is stored in the "rootX" and "rootY" attributes of the document element.

The dimensions of the preview are stored in the "originalDocumentWidth" and "originalDocumentHeight" attributes of the document element. The unit is pixels.

Each panel (except for the root panel) is attached to a specific side of its parent panel. The side index is stored in the “attachedToSide” attribute of each panel. The scheme below shows which sides correspond to the indexes.



Each attached panel is rotated depending on the side it is attached to. With side 0 it’s always attached to a side of its parent panel. The coordinate space of the panel is rotated compared to the coordinate space of its parent panel:



By default, a child panel is positioned at the center of the parent side. But it is possible it is moved, however. This offset is stored in the “hingeOffset” attribute of each panel. The unit of this value is pixel.

To create the bitmap you can use System.Drawing.Bitmap. This class can easily output to a JPG file.

Ready, set, code!

4. Bonus points

Cracked the code in a heartbeat? Too easy for you?

Get extra credits by:

- not generating an image, but a vector version in the form of a PDF file.

- adding another dimension if 2D is too boring. We even added the textures in the “extra” folder.

The bonus points are not a requirement, don’t fret it if you don’t do it, we won’t hold it against you.

5. Wow us!

That’s it – send us your finished file at work4chili@chili-publish.com.

Let us know what you think of the challenge (soft/medium/hot) and the code (soft/medium/hot).

And whether or not you think we could be the place for you to be.

Come work4chili.com. Or get in touch to develop the future of coding.