

HOW TO CALCULATE CUSTOM OFP ASPECT RATIO SETTINGS

1. Planning

Start by comparing your screen format to a 4:3 display. Find two sets of dimensions:

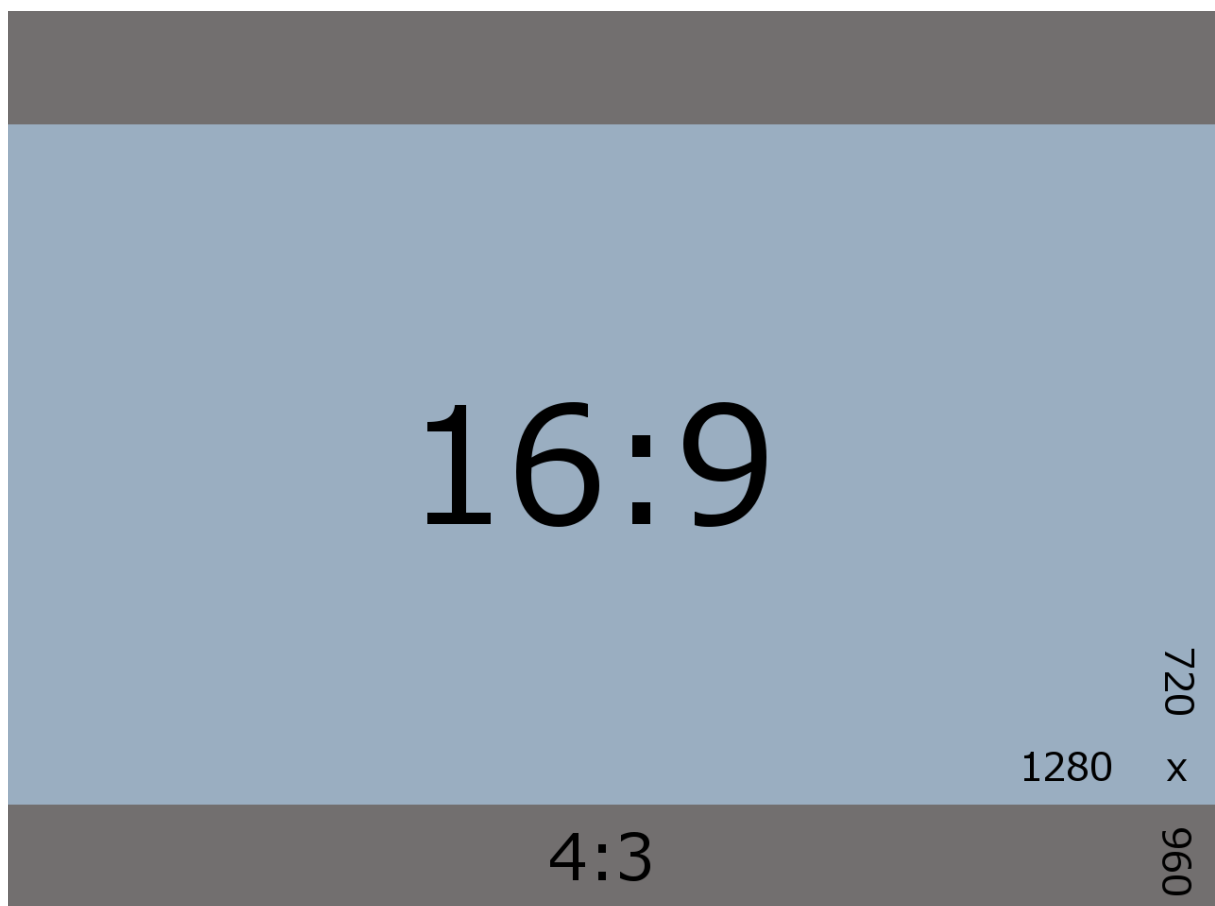
- yours and 4:3 with the same width
- yours and 4:3 with the same height

I recommend to use this calculator to find new dimensions:

http://andrew.hedges.name/experiments/aspect_ratio/

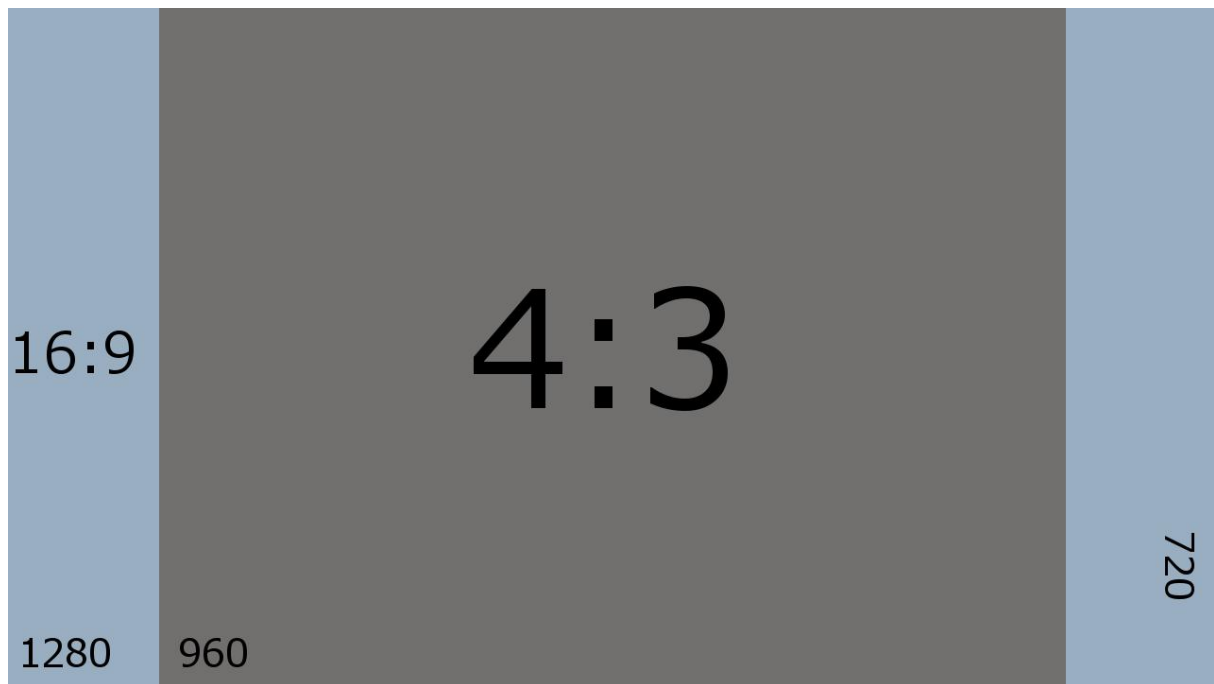
For example: I have a 16:9 monitor. One of it's frequently used dimensions is 1280x720. I'm going to compare it to a 4:3 resolution that has the same width...

1280x720 vs. **1280x960**



...and the same height:

1280x**720** vs. 960x**720**



In the first picture 16:9 image is decreased to fit inside 4:3. In the second one it's larger than the original. I want to have more screen space so I'm going with the latter. Vertical size is going to remain the same while horizontal size is going to be increased.

2. Correcting Proportions

Next step is to calculate new field of view (FOV) values. There are two numbers: horizontal and vertical size. Default FOV for 4:3 is:

```
fovLeft = 1           // horizontal
fovTop  = 0.75        // vertical
```

Here are formulas for these properties:

$$\mathbf{fovLeft} = \frac{width \times fovTop}{height}$$

$$\mathbf{fovTop} = \frac{height \times fovLeft}{width}$$

Calculate only one that needs to be modified. In my 16:9 example vertical size doesn't change therefore I only need to figure out the new `fovLeft`.

$$fovLeft = \frac{1280 \times 0.75}{720} = 1.333333$$

Round all results to six digits after decimal point.

Besides FOV we also need to get new interface (UI) position. Default for 4:3 is:

```
uiTopLeftX = 0           // horizontal
uiBottomRightX = 1        // horizontal
uiTopLeftY = 0           // vertical
uiBottomRightY = 1        // vertical
```

Formulas for interface:

$$\mathbf{uiTopLeftX} = \frac{fovLeft - fovLeft_{4:3}}{fovLeft \times 2}$$

$$\mathbf{uiBottomRightX} = \frac{\frac{fovLeft - fovLeft_{4:3}}{2} + fovLeft_{4:3}}{fovLeft}$$

$$\mathbf{uiTopLeftY} = \frac{fovTop - fovTop_{4:3}}{fovTop \times 2}$$

$$\mathbf{uiBottomRightY} = \frac{\frac{fovTop - fovTop_{4:3}}{2} + fovTop_{4:3}}{fovTop}$$

$fovLeft_{4:3}$ and $fovTop_{4:3}$ stand for default (4:3) FOV values (see previous page).

In my 16:9 example I only calculate a new horizontal position:

$$uiTopLeftX = \frac{1.333333 - 1}{1.333333 \times 2} = 0.125$$

$$uiBottomRightX = \frac{\frac{1.333333 - 1}{2} + 1}{1.333333} = 0.875$$

Open *userInfo.cfg*, write your results and test them in the game. 3D world should have correct proportions and the interface should be centered in the middle of the screen.

```
fovLeft = 1.333333;  
fovTop = 0.75;  
uiTopLeftX = 0.125;  
uiBottomRightX = 0.875;  
uiTopLeftY = 0;  
uiBottomRightY = 1;
```

3. Extending Interface

Final step is to extend the interface elements so that they use the new space. You need to calculate percentage of the extended screen. Here are formulas:

$$AR_modifX = \frac{width \times uiTopLeftX}{width - 2 \times (width \times uiTopLeftX)}$$

$$AR_modifY = \frac{height \times uiTopLeftY}{height - 2 \times (height \times uiTopLeftY)}$$

AR_modifX is horizontal size and AR_modifY is vertical size. width and height indicate your new resolution. In my 16:9 example I make the following computation:

$$AR_modifX = \frac{1280 \times 0.125}{1280 - 2 \times (1280 \times 0.125)} = 0.166667$$

Open file *Aspect_Ratio.hpp* in a text editor. Create a name for your new screen format and write it in line 5 replacing the current setting. For example:

```
#define aspect_ratio_my169
```

Find the line that says „Insert custom formats here” and write your own settings similar to this code:

```
#ifndef aspect_ratio_my169  
    #define AR_modifX 0.166667  
#endif
```

Save changes and launch the game. Gap in the interface should be gone.

APPENDIX

1. Scaling up

Let's say I want to have even wider field of view (zoom out). I simply need to multiply both FOV numbers by the same factor. For example:

$$fovLeft = 1.333333 \times 2 = 2.666667$$

$$fovTop = 0.75 \times 2 = 1.5$$

All other values do not change.

2. Portrait

For a rotated monitor reverse FOV and then calculate new interface values.

$$fovTop = 1.333333$$

$$fovLeft = 0.75$$

$$uiTopLeftX = \frac{0.75 - 1}{0.75 \times 2} = -0.166667$$

$$uiBottomRightX = \frac{\frac{0.75 - 1}{2} + 1}{0.75} = 1.166667$$

$$uiTopLeftY = \frac{1.333333 - 0.75}{1.333333 \times 2} = 0.21875$$

$$uiBottomRightY = \frac{\frac{1.333333 - 0.75}{2} + 0.75}{1.333333} = 0.78125$$

Don't forget that resolution is now 720x1280.

$$AR_{modifyX} = \frac{720 \times -0.166667}{720 - 2 \times (720 \times -0.166667)} = -0.125$$

$$AR_{modifyY} = \frac{1280 \times 0.21875}{1280 - 2 \times (1280 \times 0.21875)} = 0.388889$$

OFP Aspect Ratio package currently does not support portrait mode. I'll add that in the future.

You might want to scale up to have the same interface as in 4:3 (instead of a shrunk one). Revert horizontal FOV to the original value and then use it as basis to calculate new values:

$$fovLeft = 1$$

$$fovTop = \frac{1280 \times 1}{720} = 1.777778$$

$$uiTopLeftX = \frac{1 - 1}{1 \times 2} = 0$$

$$uiBottomRightX = \frac{\frac{1 - 1}{2} + 1}{1} = 1$$

$$uiTopLeftY = \frac{1.777778 - 0.75}{1.777778 \times 2} = 0.289063$$

$$uiBottomRightY = \frac{\frac{1.777778 - 0.75}{2} + 0.75}{1.777778} = 0.710937$$

$$AR_{modifyX} = \frac{720 \times 0}{720 - 2 \times (720 \times 0)} = 0$$

$$AR_{modifyY} = \frac{1280 \times 0.289063}{1280 - 2 \times (1280 \times 0.289063)} = 0.685188$$

CREDITS

Thanks to easySpec <https://www.youtube.com/user/easySPEC> for providing FOV and UI formulas.

Here's a link to his spreadsheet:

https://docs.google.com/spreadsheets/d/1JzXxW5cBFweU_dP2A7wxdPSmvddC0UzkJLRB_XORxuU/edit#gid=606746035Maybe