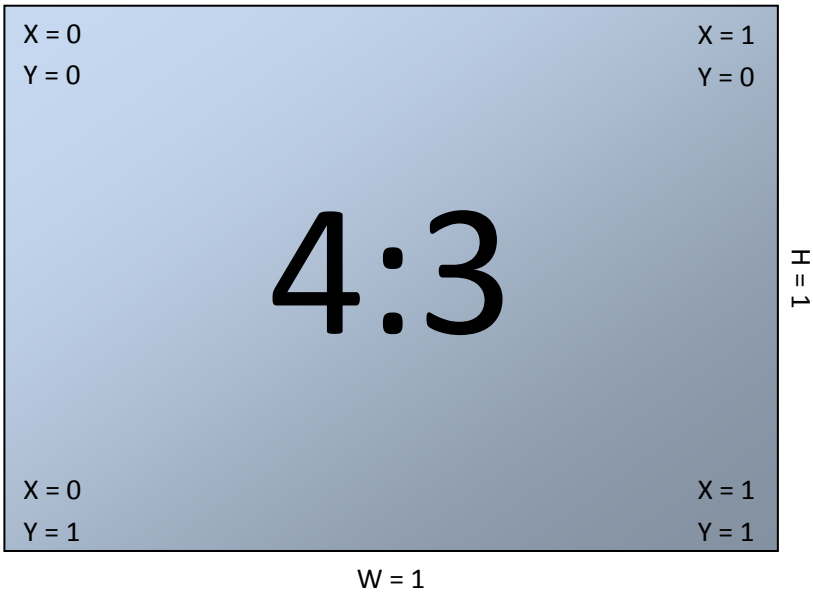


GUIDE FOR MISSIONMAKERS

1. Understanding Changes

Here's a representation of the original game dialog space:



After modifying values in the `userinfo.cfg` available area becomes larger and original space is centered.



Most of the interfaces should look fine.



crCTI build menu on 4:3



crCTI build menu on a screen with extended width and height

However, if a dialog contains background aligned with the edge of the screen then you may need to enlarge it. For example: Kegetys' Spectating Script.



spectating script on 4:3



spectating script on a screen with extended width and height

2. Modifying Description.ext

Here's a simple guideline on how to shift interface elements:

- Items in the center remain the same
- Items close to the screen edge need to be moved along with the edge
- Items that span across the screen need to be enlarged to cover the extended area

You will need **Fwatch 1.15** or newer in order to detect a different screen format.

Open *description.ext* file in text editor and paste this code at the beginning.

```
__EXEC(AR_CENTER=0; AR_modifX=0; AR_modifY=0; AR_modifX_2NDMON=0; version=0);

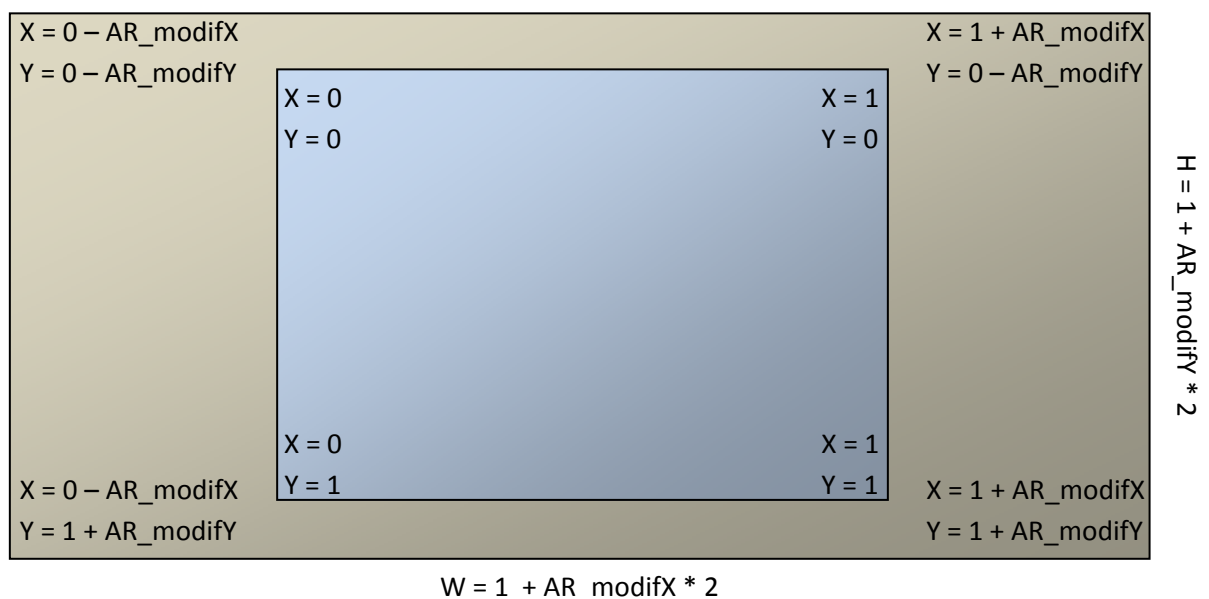
temp1 = __EVAL(if (localize "STR_USRACT_CHEAT_1" == "FWATCH") then {version = call
loadFile ":info version"});

temp2 = __EVAL(if (version>=1.15) then {call preprocessFile
"..\\fwatch\\data\\getAspectRatio.sqf"});
```

It's safe for non-Fwatch users. It initializes global variables:

- **AR_CENTER** – whether user enabled HUD centering (0 – off, 1 – on)
- **AR_modifX** – half the size of the extended horizontal space
- **AR_modifY** – half the size of the extended vertical space
- **AR_modifX_2NDMON** – side monitor size (for triplehead users)

Finding coordinates:



To move element to the right we simply need to add **AR_modifX**, to the left – subtract **AR_modifX**. To move it up – subtract **AR_modifY**. To move down – add **AR_modifY**. For example:

```
class Spectation : Title
{
    x = __EVAL(0.7 + AR_modifX);
    y = __EVAL(0.94 + AR_modifY);
};
```

__EVAL() stands for „evaluate”. It’s a command that executes given expression and returns a value to the class property. Scripting commands and variables can be written inside the parenthesis.

To make a background cover the entire screen we need to add twice the **AR_modifX** or twice the **AR_modifY** to its width or height. For example:

```
class Background4 : MSStext
{
    w = __EVAL(1 + AR_modifX * 2);
};
```

The result:



fixed spectating script dialog

As you can see interface elements are now closer to the screen edge. It will look similar on the other screen formats as well.

To view entire code download **Modified Spectating Script**:

http://ofp-faguss.com/modified_spectating_script

For a simple example (and a testing convenience) see `__AspectRatio.Intro` mission from the **Fwatch Practical Examples** package:

http://ofp-faguss.com/fwatch/examples/fwatch_practical_examples.rar