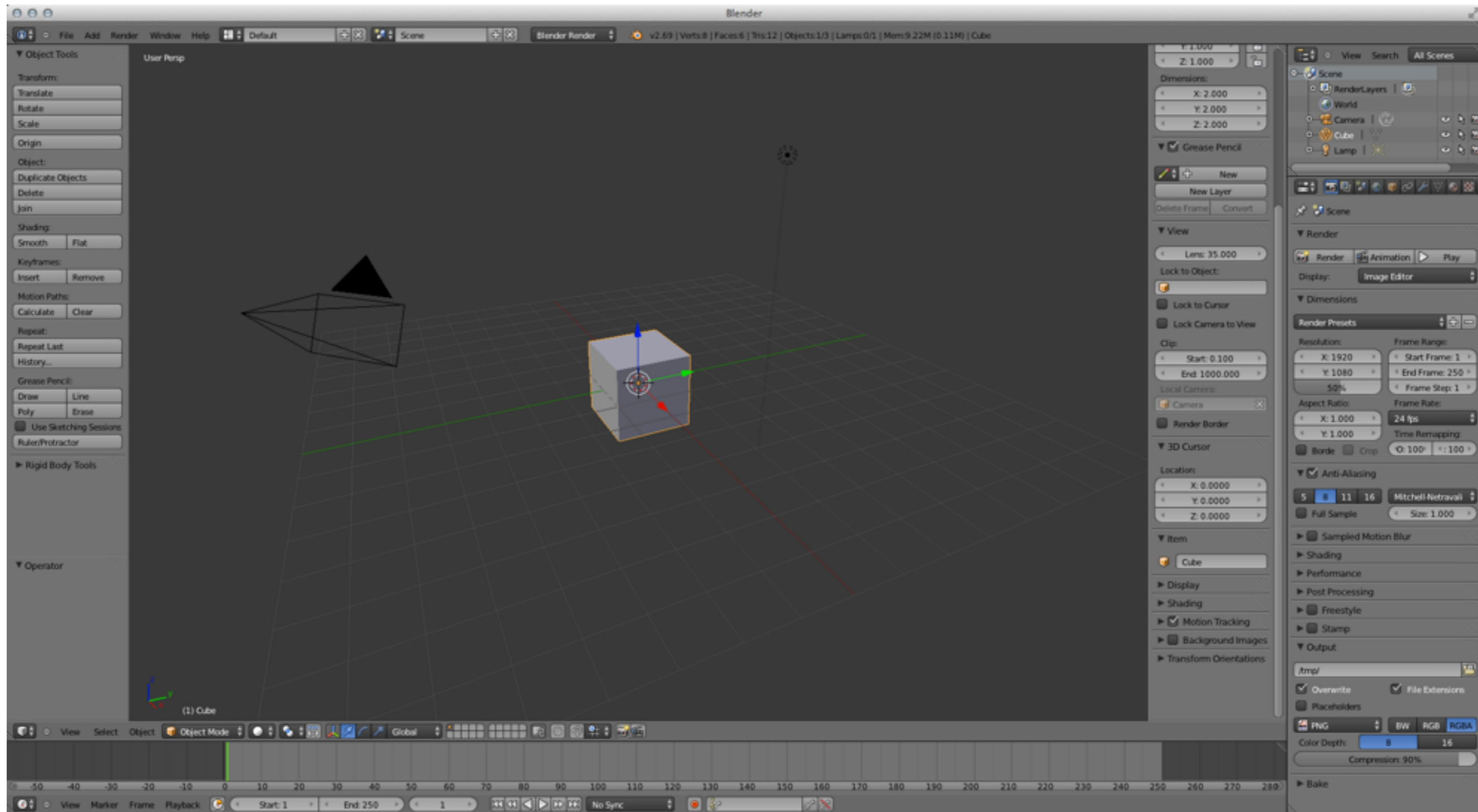


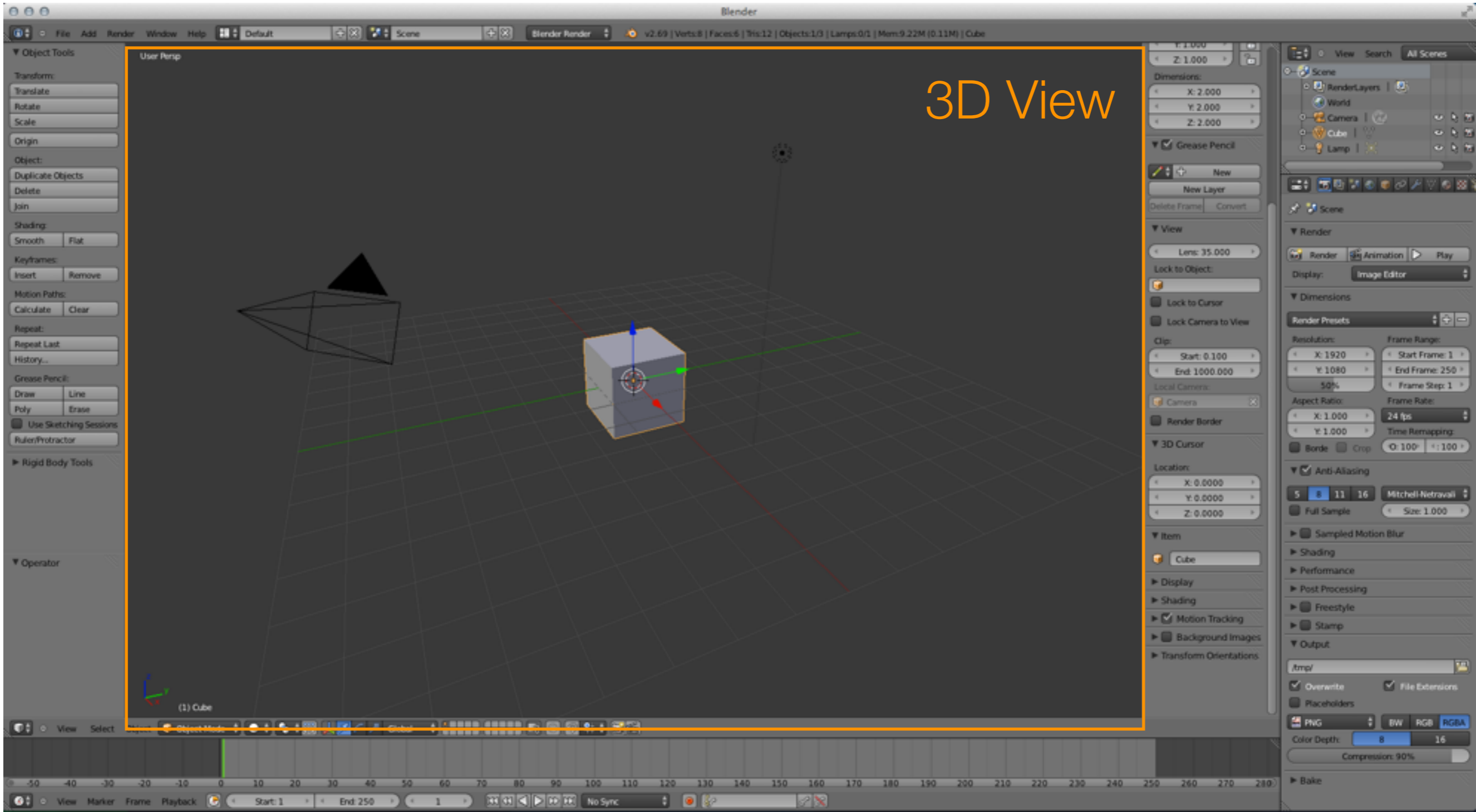
Blender

What is Blender exactly?

- Blender is OpenSource 3D package
 - This means that it provides full pipeline for 3D production
 - Modelling
 - Rendering
 - Animation
 - Postproduction
- Blender is intended for expert user, somewhat steep-learning curve
- Really need a 3 button mouse!

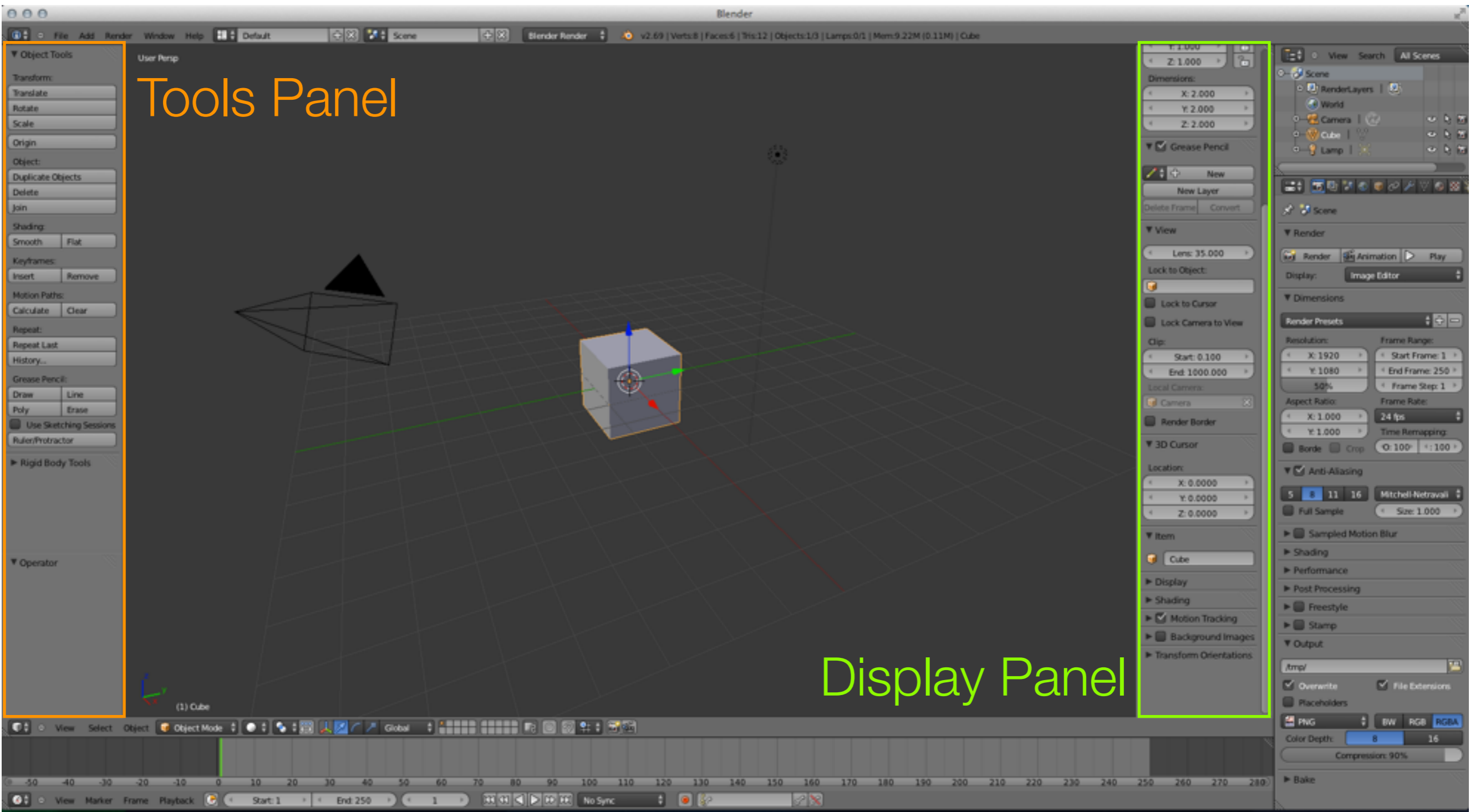
Keeping things simple!

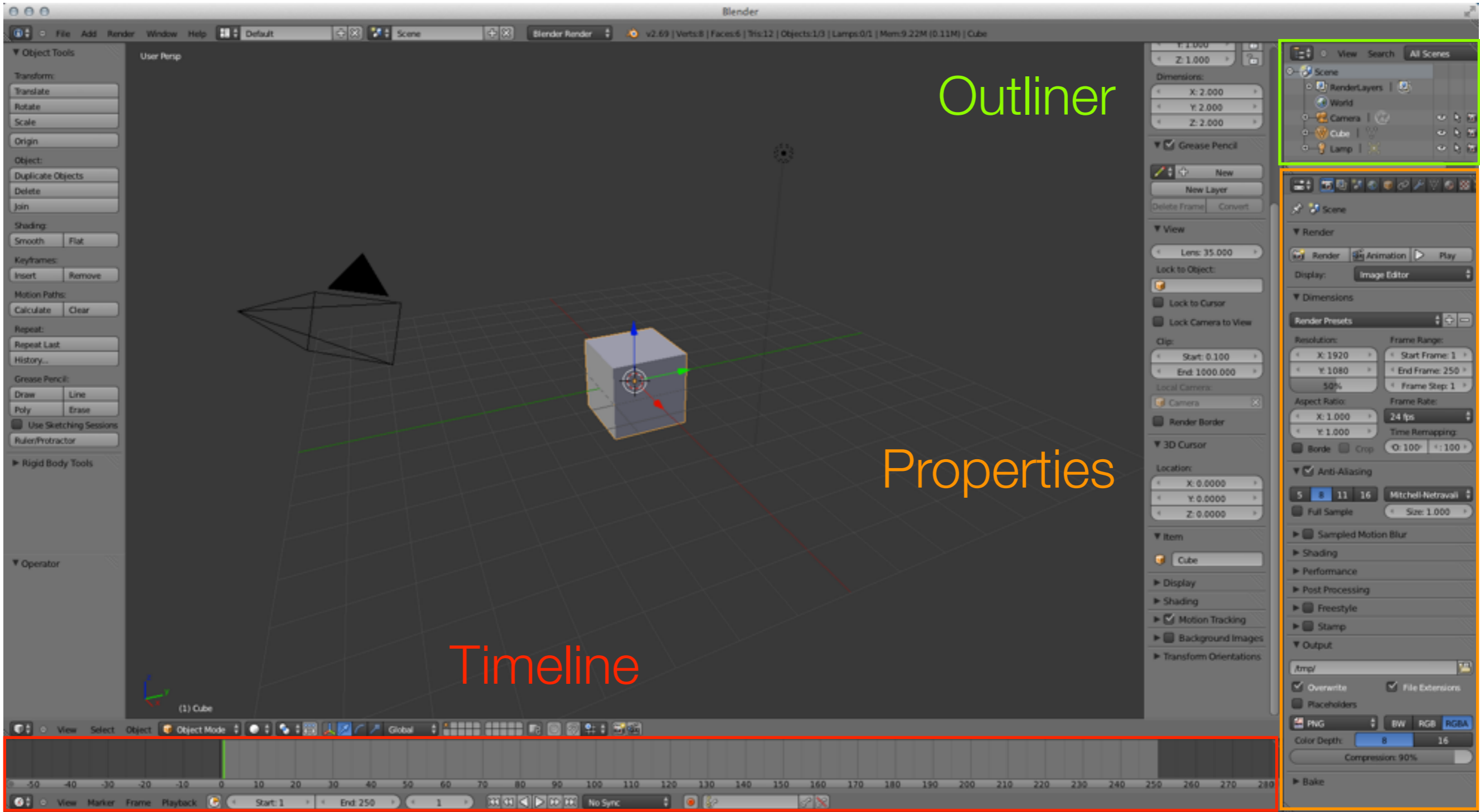




Tools Panel

Display Panel



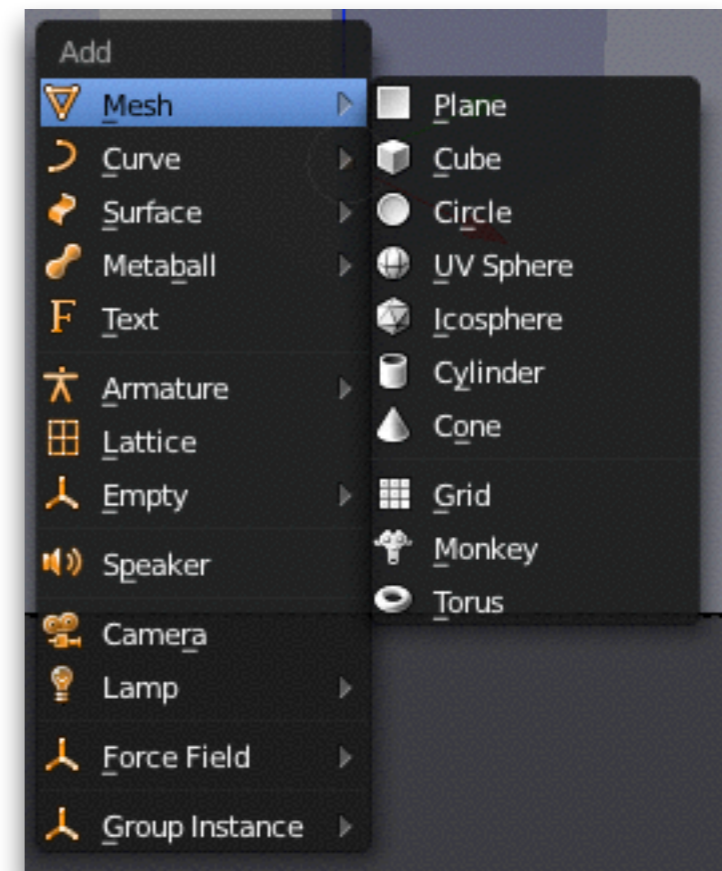


Navigation

- Zoom - Mouse Wheel / **Ctrl** + Middle Mouse Click + drag
- Pan - **Shift** + Middle Mouse Click + drag
- Rotation - Middle Mouse Click + drag
- Takes little practice before you will get comfortable with it.

Object Mode

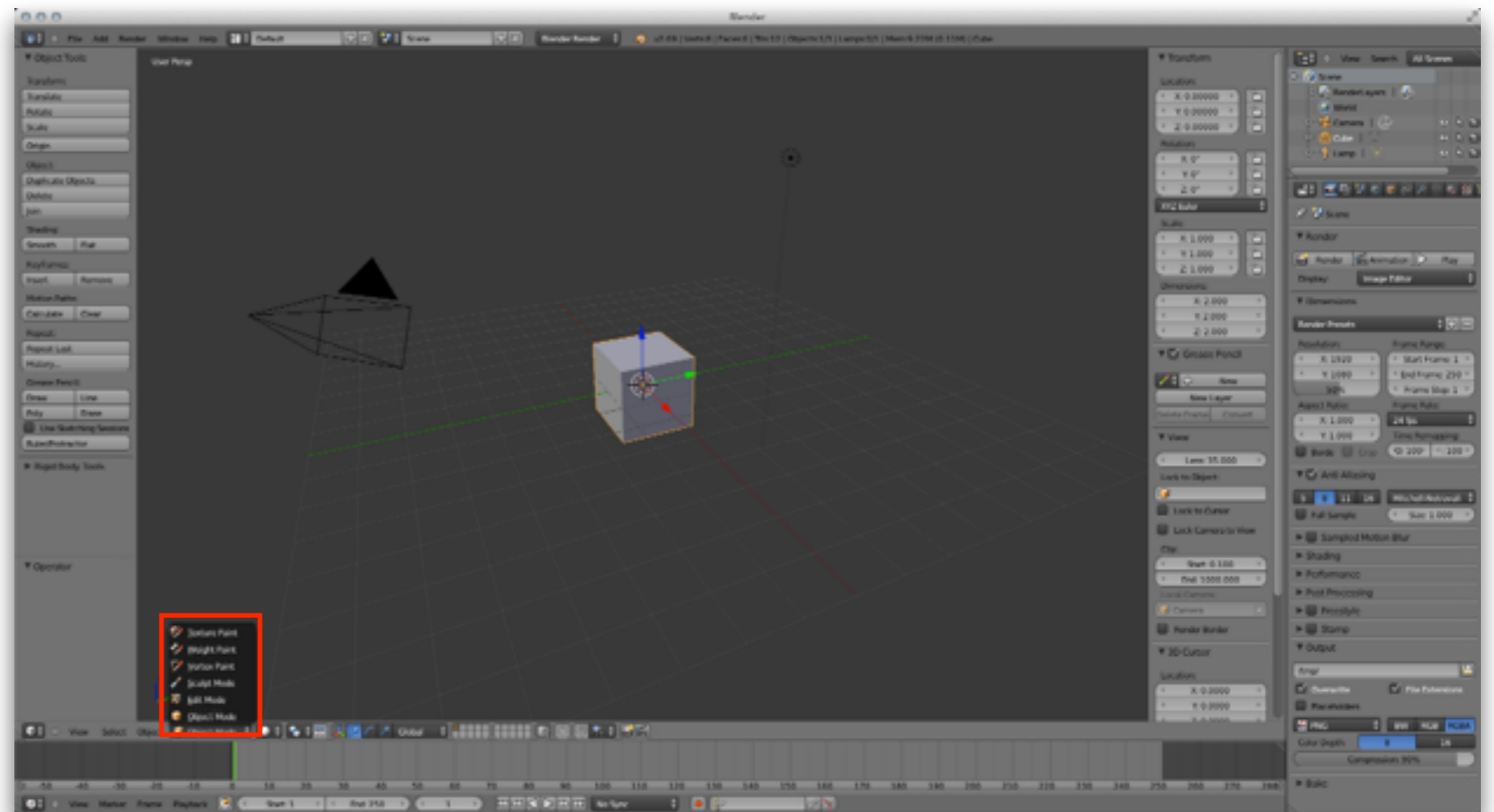
- Adding new objects to the scene
- **Shift + A**
- Also can import/export
 - **File -> Import -> ...**



Object Editing

- Blender 3D View has multiple working modes

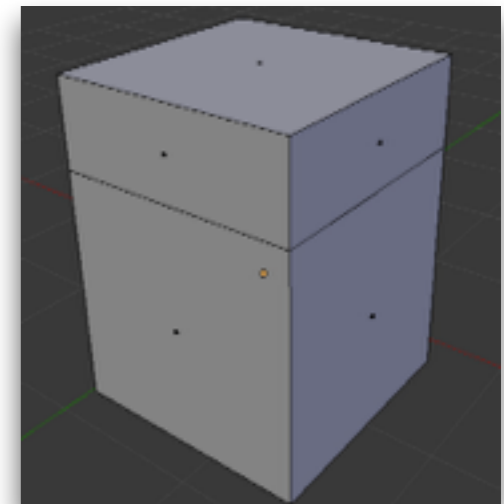
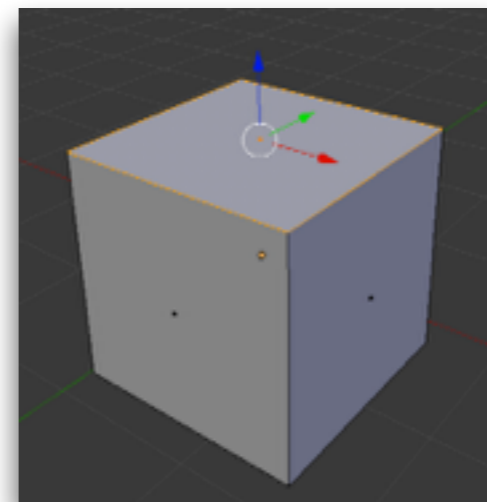
- Object Mode
- Edit Mode
- Sculpt Mode
- Vertex Paint
- Weight Paint
- Texture Paint



- Mesh editing is done in Edit mode (switch with Tab)

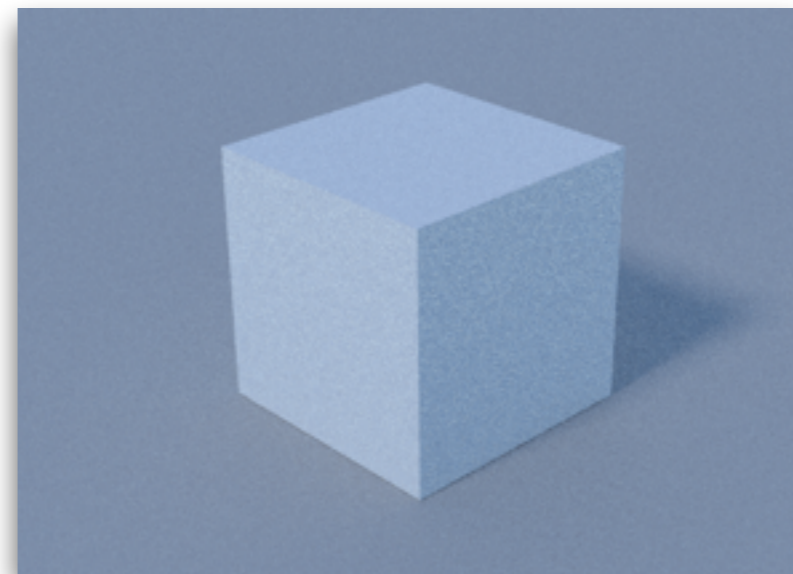
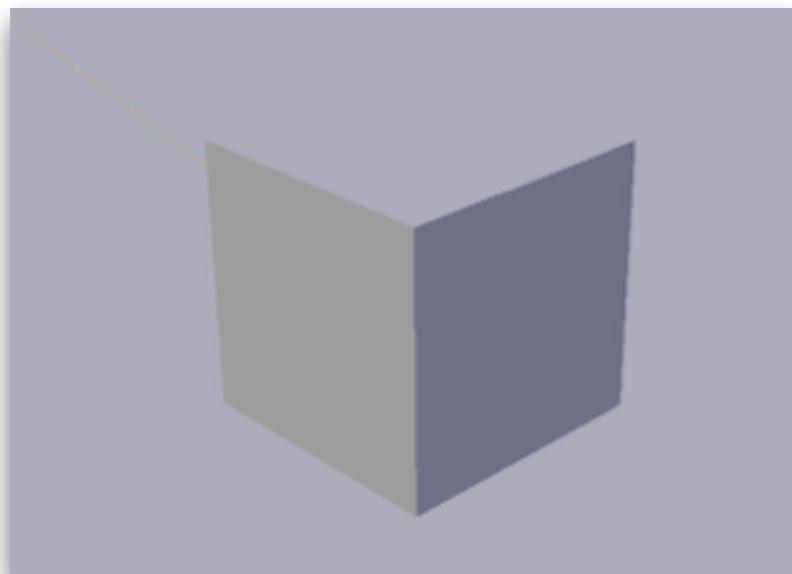
Edit Mode

- Different selection modes Vertex / Edge / Face
- Right Mouse Click to select an element
 - Can also do Rectangle (**B**) / Circle selection (**C**)
- Basic Manipulation
 - **R** - Rotate
 - **G** - Translate (grab)
 - **S** - Scale
- Once elements are selected there is a wide range of editing tools
 - **E** extrudes currently selected element
 - **Alt + M** merges elements
 - **Ctrl + E** menu, **W** menu
 - **Ctrl + R** creates edge loop



Rendering

- Rendering is a large topic
- Blender has 3 rendering engines - select in top panel
 - Blender Internal
 - Cycles
 - OpenGL
- For producing nice looking images overnight you would use Cycles



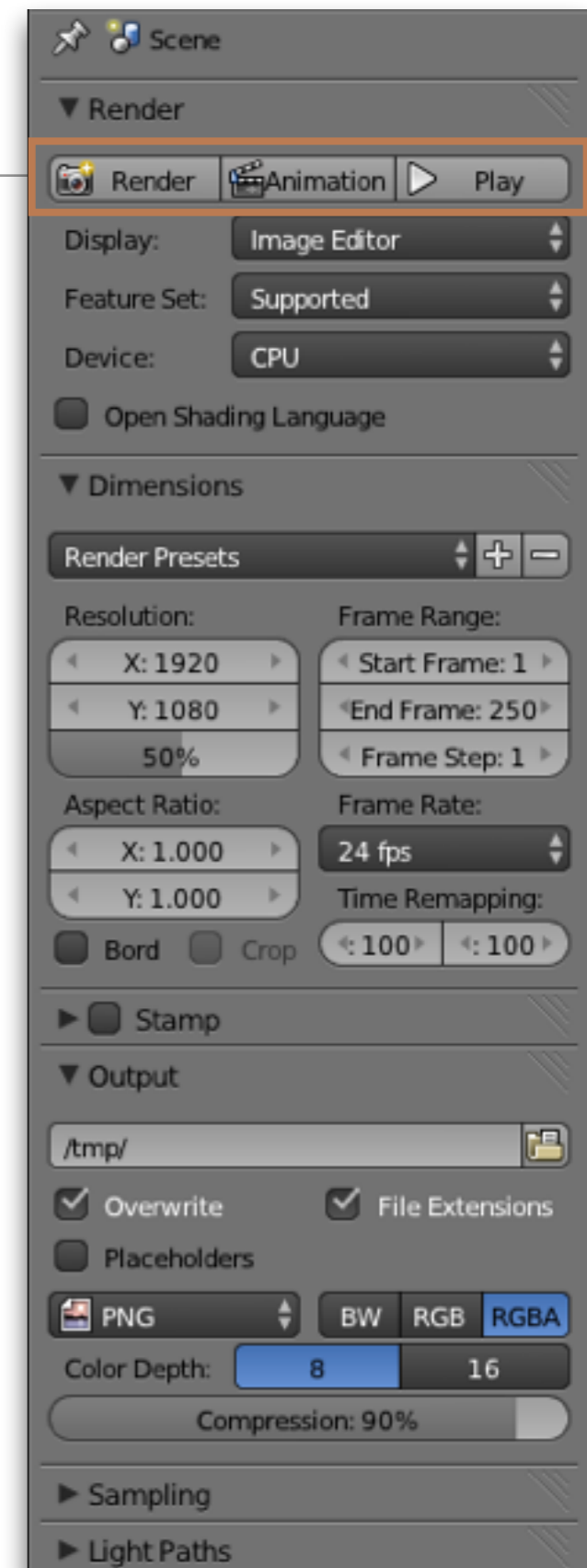
Rendering

- Requires some setup
- Adding lights (**Shift + A** -> Lamps)
 - Point
 - Sun
 - Area
 - Spot
- Setting up Sky Texture
 - Properties -> World



Render properties

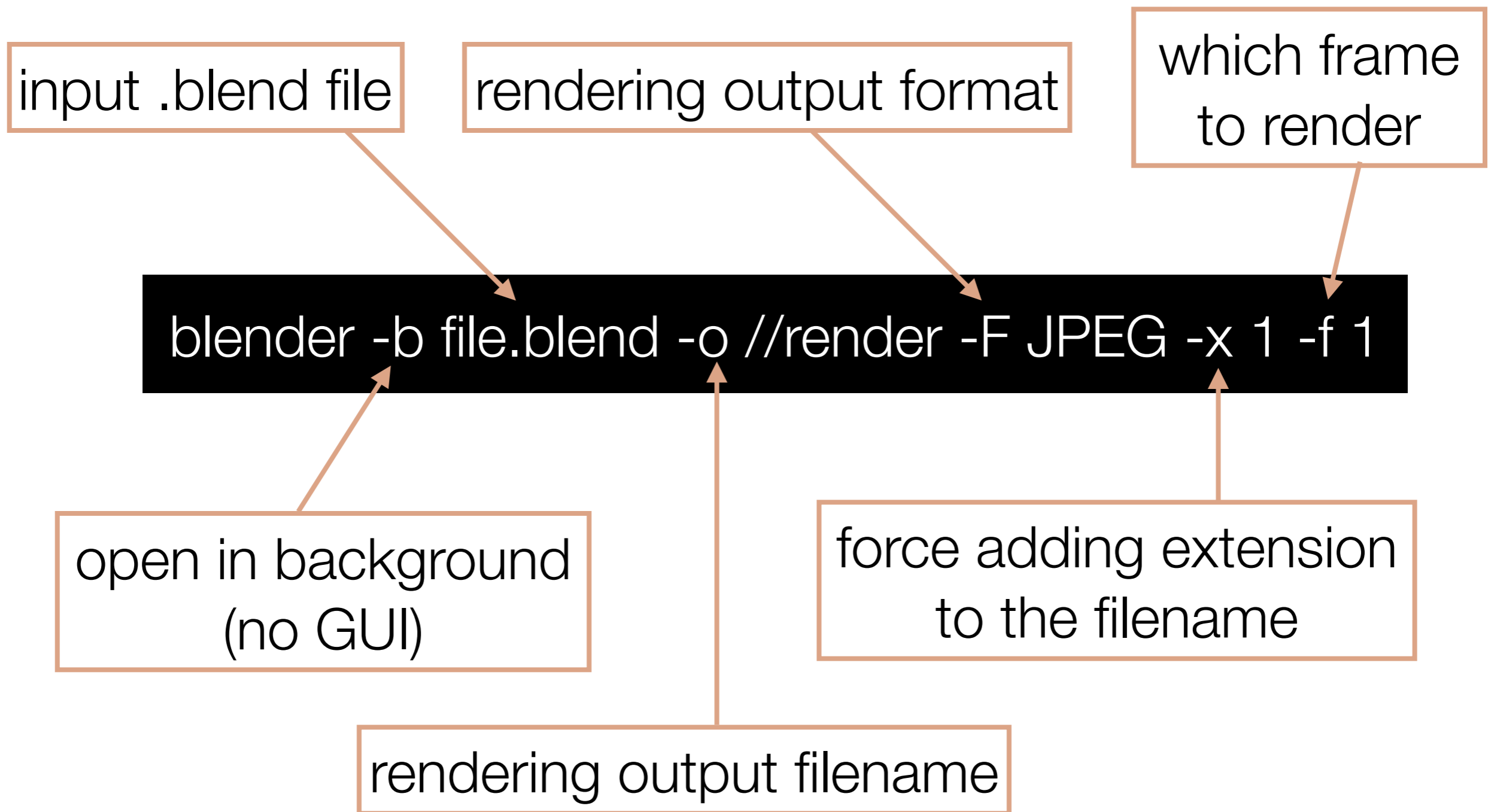
- Properties -> Render
- Lots of options
 - Output render size
 - Format
 - Path-tracer options
 - Lots of experimentation to do!



Command Line Tools and Blender API

- Blender has some nice functionalities for offline rendering
 - Requires prepared .blend file
- Blender is also easily extended by add-ons, which are written in Python
- Command line tools allow us to do many things, and we can also extend those by passing Python scripts
- Remember to setup envVar on Windows, or add alias to your .bash_profile for Mac OSX
- Good resources : [link](#)

Simple rendering



Scenario

- Huge library of meshes
- Want to have a good quality snapshot of each in the same style
- We can prepare a .blend file with lighting and camera set up
- Then use Blender command line with mesh loading script
 - `-P <filename>` - execute python script on load
- Need to invoke rendering in the script
- Can easily write bash that will be invoking the blender with our script for different files

Python script for Importing/Rendering

```
import bpy
import sys

argv = sys.argv
argv = argv[argv.index("--") + 1:] # get all args after "--"

# parse args
obj_in = argv[0]
render_out = argv[1]
format_out = argv[2]

# import obj
bpy.ops.import_scene.obj(filepath=obj_in, axis_forward='-Z', axis_up='Y')

# set output filename
bpy.data.scenes["Scene"].render.image_settings.file_format = format_out
bpy.data.scenes["Scene"].render.filepath = render_out

# render
bpy.ops.render.render(write_still=True)
```

`blender -b file.blend -import_render.py -- file.obj render PNG`

Nice scripting resource : <http://blendscripting.blogspot.com>

Thanks!