Hello GL!

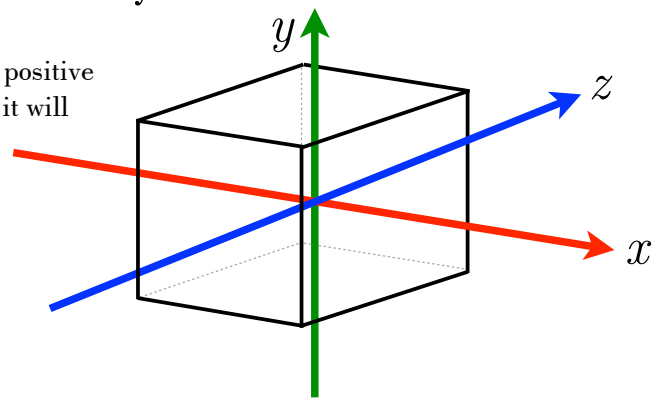
- Modern OpenGL or WebGL graphics programs have a lot going on, and need more of an introduction that just "here's a `canvas`, now draw on it"
- Some potential complicating factors outside of the graphics pipeline itself: portability (you can write GL programs in virtually any language and operating system); specialization (a whole new language, GLSL, is involved—but is fortunately the same regardless of the platform)

The Default Space

- It might surprise you to learn that your 3D environment measures 2×2×2, with (0, 0, 0) at the center and bounds of [–1, 1] for every axis:

(arrow heads indicate the positive direction—but take note, it will change for *z* later on)

What you see onscreen is the view on the near plane, the 2×2 square defined by *z*=–1



- But, like the TARDIS, it's bigger than it seems!

Parts and Flow

- The diagram below describes the general code base and execution sequence of the kinds of 3D graphics programs we'll be writing (yes, that is available in full-page format also)
- Crossed out portions indicate portions that are not yet included in introductory sample code
- Shaders are written in GLSL, drawing and graphics initialization uses GL calls, and everything else is in the "native" language of the program