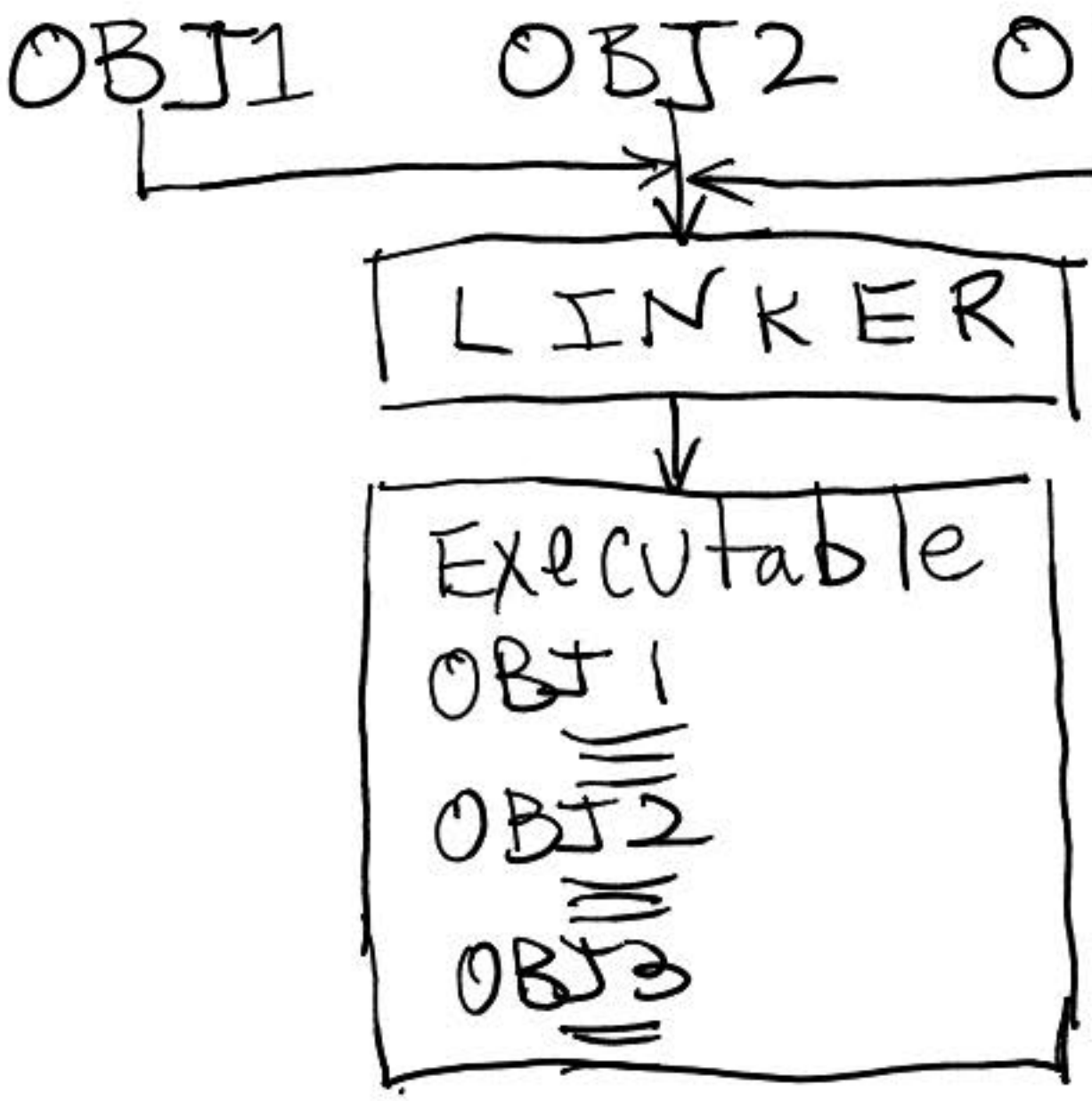
Dynamic Linking

We learned about static Linking



In static linking, what is linked statically gets dumped into the executable.

What about using a HUGE library?

Every executable using it would be HUGE too!

How about using a shared library?

Sure why not. We link to it (mostly) at run time

Dynamic Linking.

Executable

```
main()
  ≡
  ≡
etc()
  ≡
  ≡
Foo()   // stub
```

if shared library in which "Real" Foo() lives has not been loaded:

Try loading it
If failure, die.

If real Foo() has not been found:

Try to find it
If failure, die

Branch to real Foo()

Shared library

TOC:

Real Foo() → Real Foo() {
≡
≡
}

Dynamic Linking

On windows

* .lib has stubs
* .dll has "real" code

.dll means "dynamically (loaded / linked) Library"