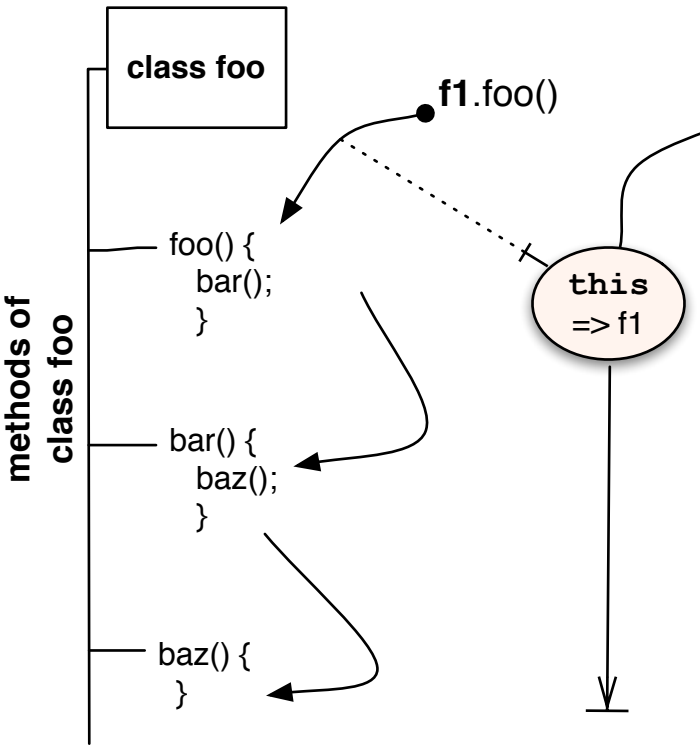
"this" propogation in Java and Javascript

hursh jain/mollypages.org

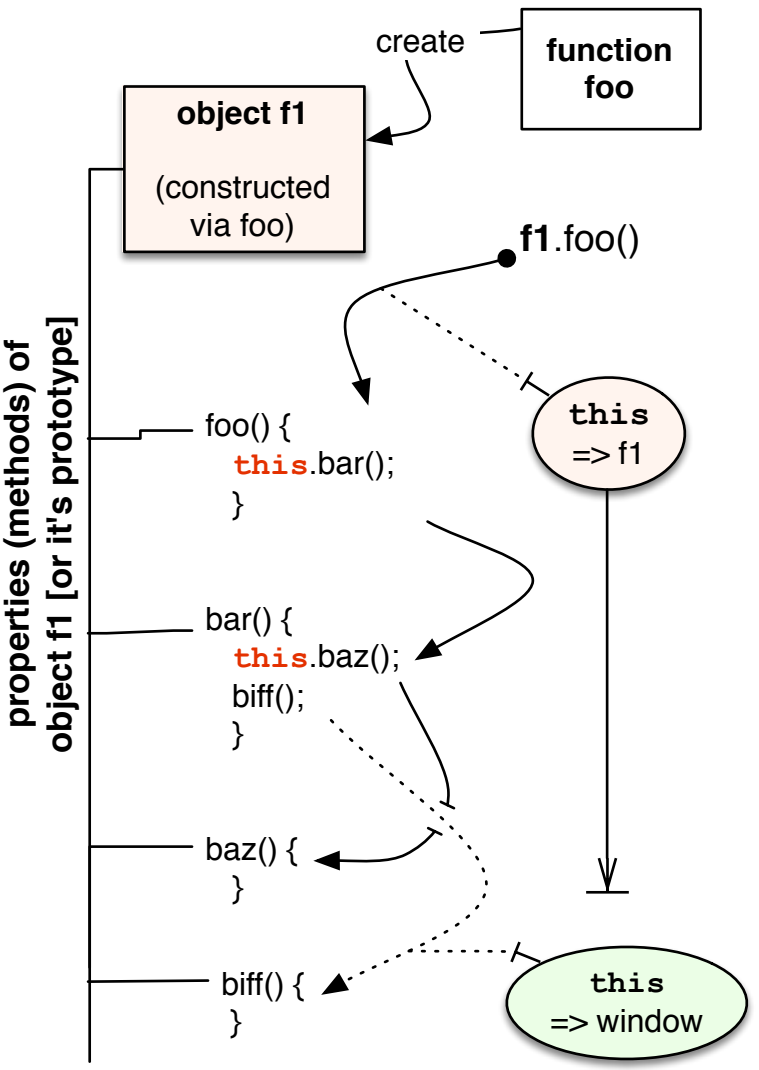


Java

`this` is automatically propogated.

We DO NOT HAVE to say `this`.bar() and `this`.baz() to keep track of `this` down the line, in those further methods. (although we could say it, if we wanted to, but not needed).

Static methods are invoked on the class (not object) and have no `this` pointer at all. Note, in Java, methods belong to the class. In JS, methods belong to an object (or it's prototype).



JavaScript

this is NOT automatically propogated, even when bar and baz are properties/ functions of the instance f1

We DO HAVE to say this.bar() and this.baz() to keep track of this down the line, in those further methods.

Since, biff() is invoked without an explicit `this`, `this` refers to the window and not to f1 inside biff (even though biff is a property/method of the object f1).

`this` **always(*)** refers to the *original* object through which the method was invoked (this can be easy to loose track in JS). In this example, object **f1**.

(*) the only exception is the JS5 "bind" method which can permanently change this to another object