

## Notes - 5/20: SENT

Thursday, June 09, 2011 10:17 PM

Mixins:

- Mixins
- Can mixins

- Goals
  - Completely delight current community of JS developers (live)
  - Simple and elegant design
- Possible answer:
  - No requirements
  - Only have access to your state
- Current implementation
  - Classes that you multiply inherit from

Ideas:

```
trait EventTarget { // NOish
  var listeners = []
  function ael(...) { ... }
  function de(...) {
    this.asdsadsa
  }
}
```

:!Addable where this : IEventable // NO

```
trait EventTarget extends ITargettable
class Button extends EventTarget
```

```
class EventTarget
class Button extends Element with EventTarget
```

1. Start with baseless classes
2. Use a name like 'mixin'
- 3.

```
trait Printable(...) extends Control // <- no paramlist
class Control
class Button extends Control with Printable
```

```
trait ColorPrintable(..) extends Printable
```

```
class Page extends Control with Printable, ColorPrintable
```

```
trait Printable(z) : Control, Ability
class Button(x,y,z) : Control(x,y), Printable(z)
```

private is internal

What does it compile to:

- A class – with no actual base

- Mixin composition
- Walk it's prototype and copy onto

Still something to be said for:

- Extends and implements, with and requires

Next steps;

- Strada in Strada
- Self host

Null:

- Error to infer the null type

More backlog:

- Double var declarations

**From:** Luke Hoban

**Sent:** Friday, May 20, 2011 8:43 AM

**To:** Strada Design Team

**Subject:** Strada Design Meeting Agenda - 5/20/2011

[Note – we'll be starting 30 mins late today at 10:30]

**Agenda:**

- Update on implementation from Steve
- Mixins (Thread + Mads' notes attached)
- Class parameters (Mads' notes attached + commentary from es-discuss)
- Backlog:
  - Top level scope?
  - Uninitialized variables/inference/defaults
  - O[x] with typed objects
  - Enums
  - Global module
  - Type compatibility rules
  - Foreach
  - Class-parameters as privates and inheritance
  - Function type compatibility (omit parameters, more parameters, varargs, etc.)
  - Any -> number (what gets emitted?)
- Anything else?

Luke