

Get your money's worth with Selective Functors

Jeremie Dimino, Jane Street



@diml



@dimenix

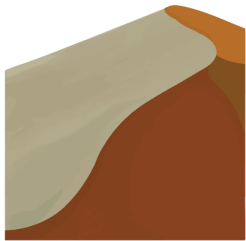


**Jane
Street**



Jane Street

OPEN SOURCE



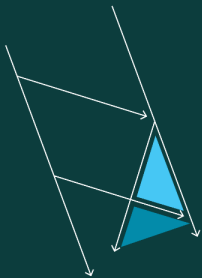
DUNE



BASE



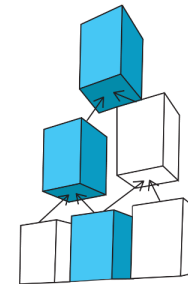
CORE



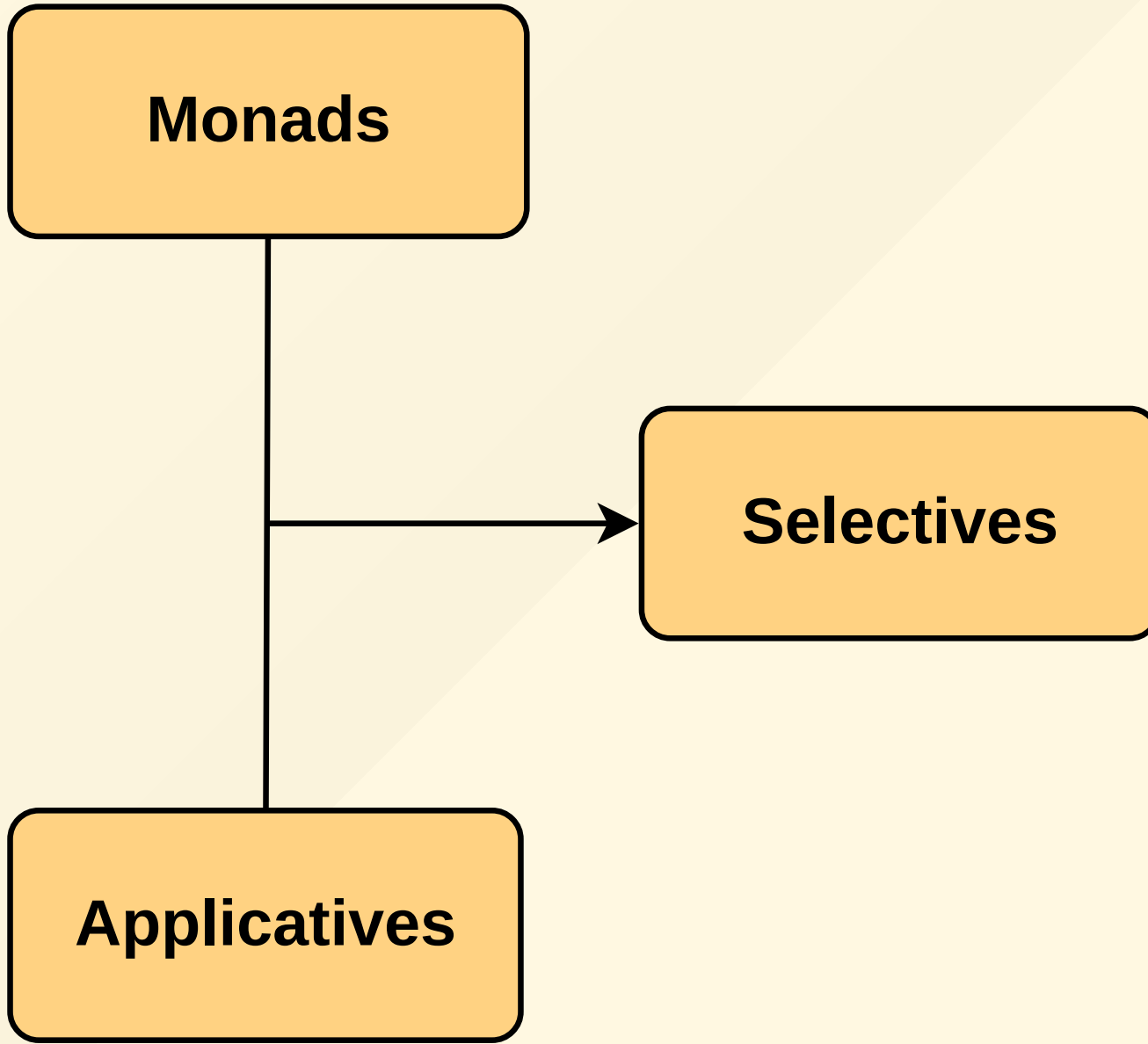
ASYNC





INCR_DOM



INCREMENTAL



github.com/janestreet

-  base
-  core
-  async
-  incr_dom
-  incremental
- ...

Over 100 packages

MONOREPO

incr_dom

incremental

async

base.caml

base

core

src/dune:

```
(library
  (public_name mylib)
  (libraries re lwt))

(rule (with-stdout-to m.ml (run gen/gen.exe)))
```

src/gen/dune:

```
(executable
  (name gen)
  (libraries ppplib))
```

Dune's internals

1. Generate rules
2. Run the build



The 'a Build.t selective

```
type rule = Action.t Build.t
```

```
val dyn_deps : ('a * Dep.Set.t) t -> 'a t  
val      deps : 'a t * Dep.Set.t    -> 'a t
```

Selective parsers

jobjo.github.io/2019/05/19/applicative-parsing.html

The end



discuss.ocaml.org



opensource.janestreet.com