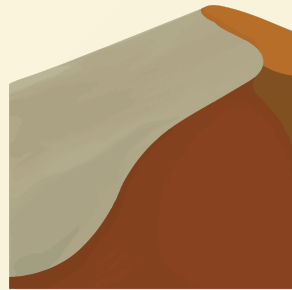


Dune

The OCaml build system



Jeremie Dimino, Jane Street

 @diml

 @dimenix

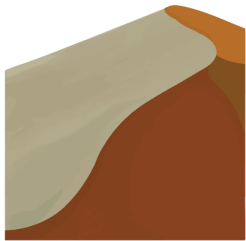


**Jane
Street**



Jane Street

OPEN SOURCE



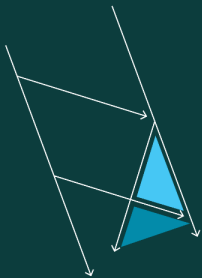
DUNE



BASE



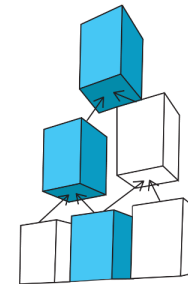
CORE



ASYNC



INCR_DOM



INCREMENTAL

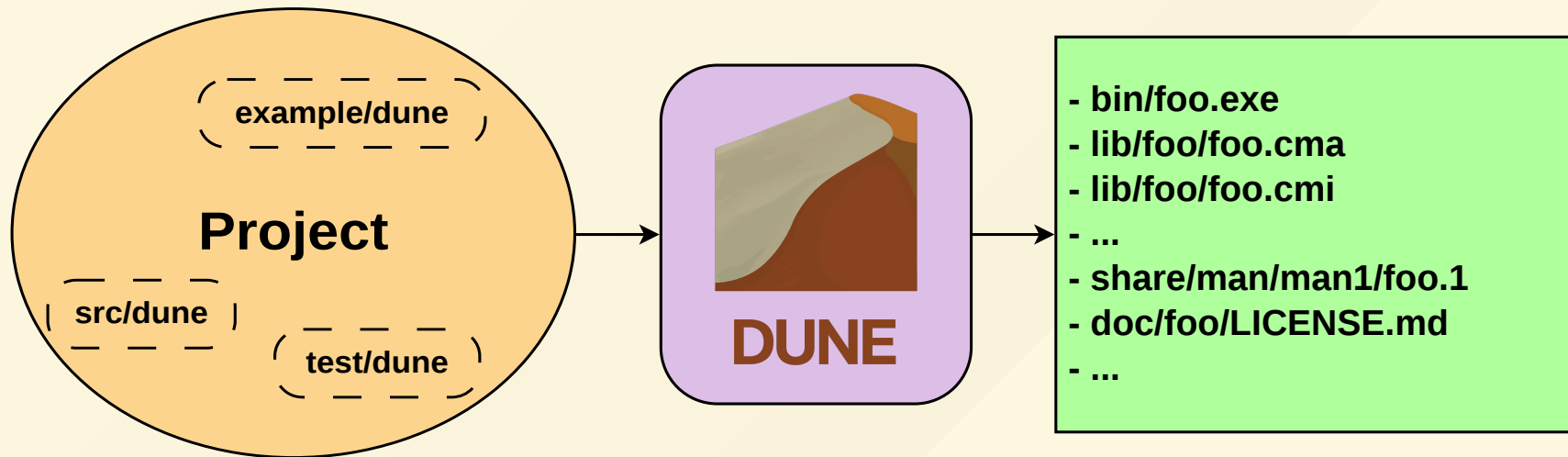
The history of Dune

At the beginning, there was Jenga and the Jenga rules...

... then a domain specific tool called jbuilder...

... and finally Dune.

What is Dune?



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

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

Dune language

```
graph TD; A[Dune language] --> B[OCaml Build Logic]; B --> C[Build system];
```

The diagram illustrates a three-step process. It begins with a solid green rounded rectangle labeled 'Dune language'. A downward arrow points from this box to a yellow rounded rectangle with a dashed border labeled 'OCaml Build Logic'. Another downward arrow points from the yellow box to an orange rounded rectangle with a dashed border labeled 'Build system'.

OCaml Build Logic

Build system

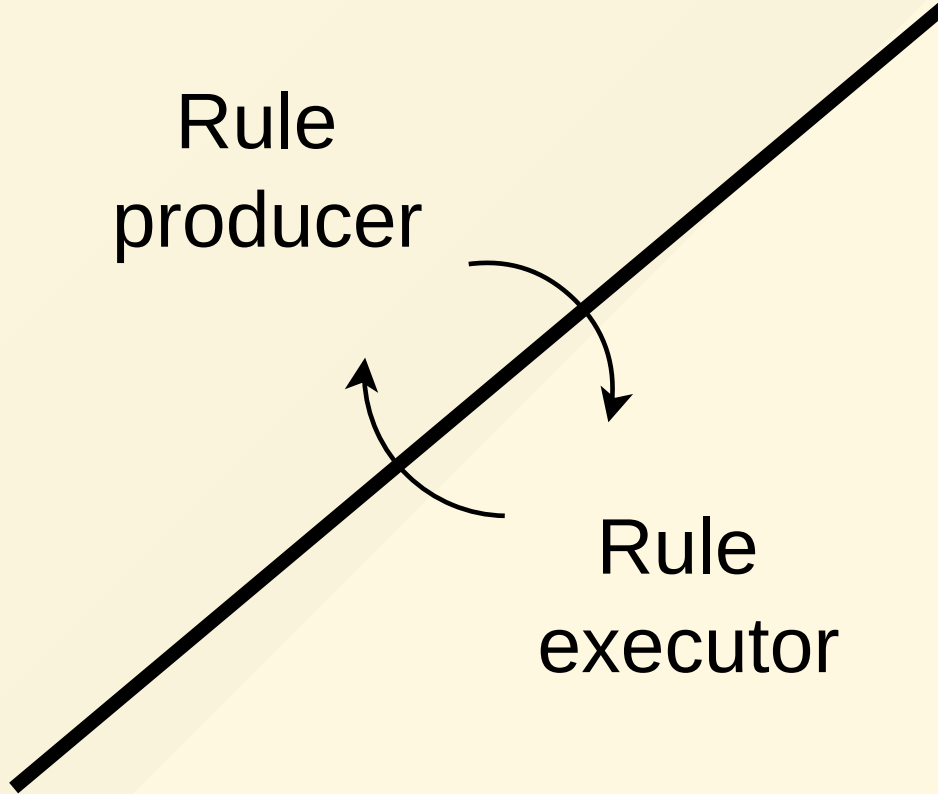
Interesting Dune features

- Composability
- Multiple build contexts

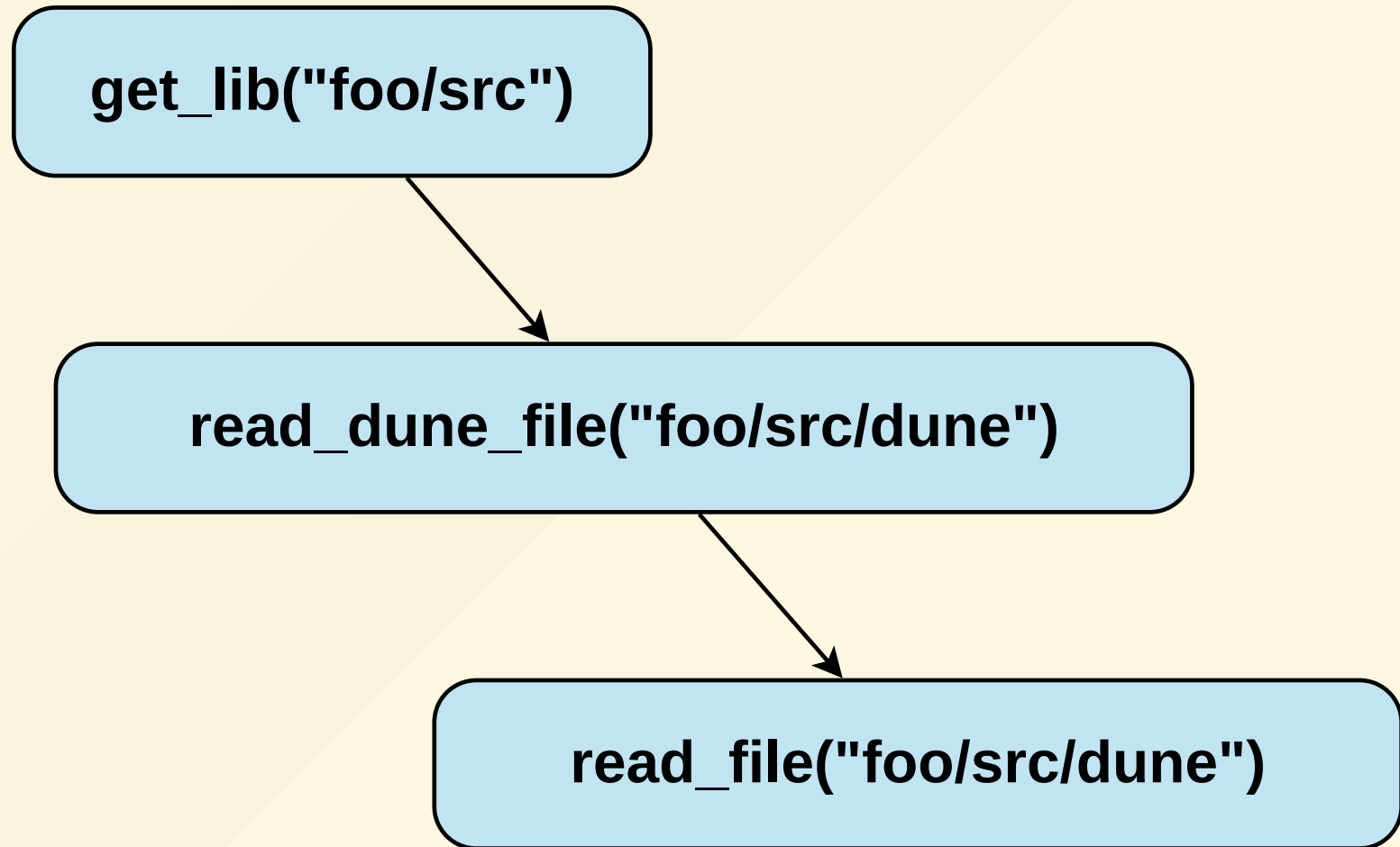
How does it work?

Rule
producer

Rule
executor



Memoisation framework



The end



dune.build



discuss.ocaml.org



opensource.janestreet.com