

```
1 // Generated by Melange
```

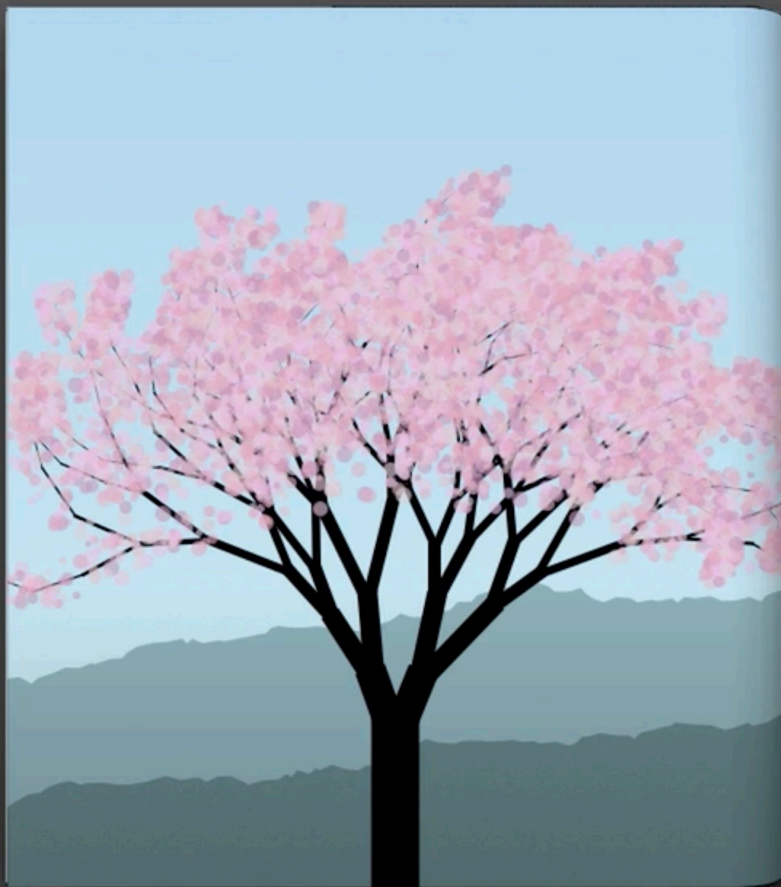
```
4 console.log("Hello, Fun OCaml!");
```

A Melange of Tooling

```
6 export {
```

```
8 }
```

```
9 /* Not a pure module  @anmonteiro
```



```
ctx.fillStyle = "#000";
ctx.fillRect(0, -width/2, length, width);

ctx.restore();

var tipX = x + (length - width/2) * Math.cos(angle);
var tipY = y + (length - width/2) * Math.sin(angle);

if (i > 4) {
  blossomPoints.push([x,y,tipX,tipY]);
}

if (i < 6) {
  drawBranches(i + 1, angle + random(-0.15, -0.05) * Math.PI);
  drawBranches(i + 1, angle + random( 0.15,  0.05) * Math.PI);
}
else if (i < 12) {
  drawBranches(i + 1, angle + random( 0.25, -0.05) * Math.PI);
}
}

function drawBlossoms (blossomPoints) {
  var colors = ["#f5ceea", "#e8d9e4", "#f7c9f3", "#ebb4cc", "#f0e68c"];
  ctx.globalAlpha = 0.60;

  for (var i = 0; i < blossomPoints.length; i++) {
    var p = blossomPoints[i];
    for (var j = 0; j < 16; j++) {
      var x = lerp(p[0], p[2], random(0,1)) + random(-10,10);
      var y = lerp(p[1], p[3], random(0,1)) + random(-10,10);

      ctx.fillStyle = colors[Math.floor(random(0,colors.length))];
      ctx.fillCircle(x, y, random(2,5));
    }
  }

  ctx.globalAlpha = 1.0;
}
```

What's my principle?

What's my principle?

Let's take a small step back in time...

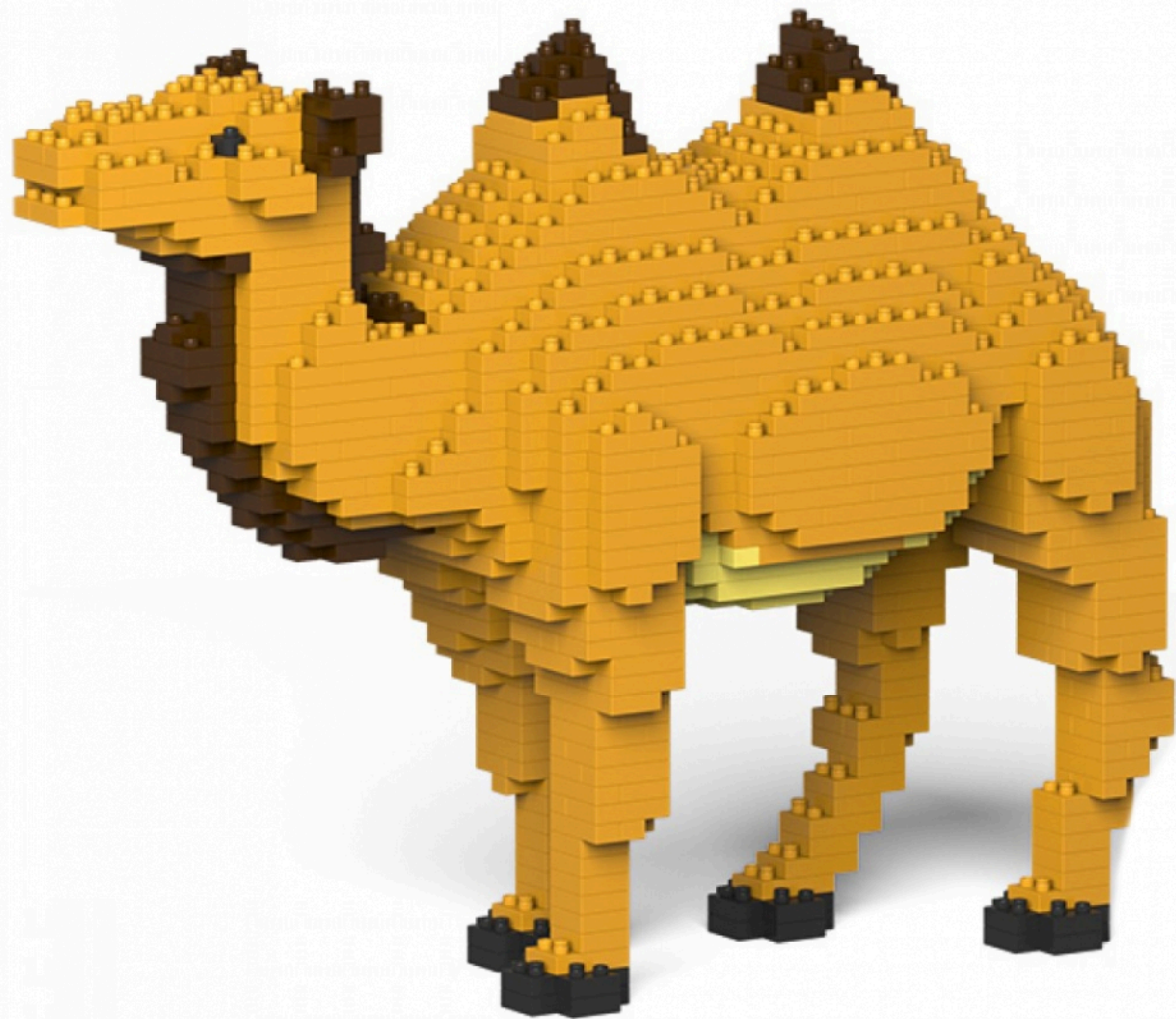


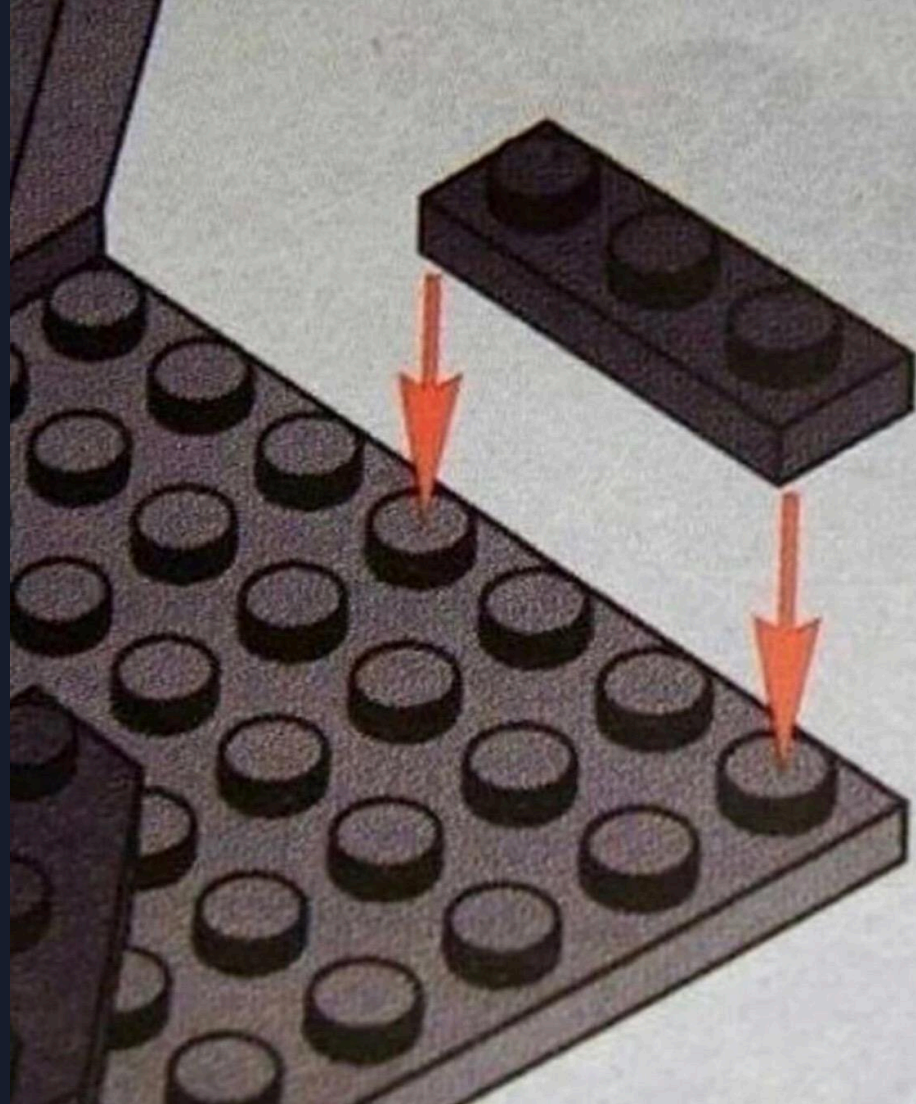
OCaml

Y **Hacker News** [new](#) | [past](#) | [comments](#) | [ask](#) | [show](#) | [jobs](#)

▲ Reason: A new interface to OCaml (facebook.github.io)

603 points by clayallsopp on [May 17, 2016](#) | [hide](#) | [past](#) | [favorite](#) | [287 comments](#)





SAINT DOGBERT SEEKS OUT
TECHNOLOGY THAT HAS BEEN
POSSESSED BY THE DEMONS
OF STUPIDITY.



S. Adams E-Mail: SCOTTADAMS@AOL.COM

HE HAPPENS ACROSS A
SOFTWARE DEVELOPER.

I'LL MAKE THE COMMAND
EASY TO REMEMBER,
LIKE "CTRL-ALT-F4-DEL".



© 1994 United Feature Syndicate, Inc.

AND IF THEY FORGET
THAT THEY CAN JUST
EDIT THE SOURCE CODE
IN "COMMAND.COM"

PERFECT.





On OCaml and the JS platform

12 Mar 2021

ReScript, née **BuckleScript**, is a state-of-the-art compiler that used to target OCaml (and Reason), but is fast moving away from its parent language. OCaml compatibility is high on my list of desirable features, and I've been disillusioned with the direction the project has veered into. So I took matters into my own hands. Read on!



Melange

OCaml for JavaScript developers.

[Get started](#)[Playground](#)

A Solid Type System

Melange leverages OCaml's powerful type system to catch more bugs at compile time.

First-Class Editor and Tooling

Melange fully utilizes the power of the OCaml Platform to

JavaScript Integration

Use existing JavaScript packages from NPM, or your own JavaScript

Stable and Industry Backed

Melange builds on top of decades of type system research,

Can OCaml Reach The Browser?



youtu.be/vSSpbAl_-t0



```
$ opam install melange
```

```
$ opam install melange
```

```
$ opam install reason-react
```

```
$ opam install melange-{json,fetch}
```

```
$ echo '(using melange 0.1)' >> dune-project
```

```
$ dune build @melange
```



Ppxlib

Standard library for PPX rewriters



Merlin

Editor helper, provides completion, typing, and source browsing in Vim and Emacs



Dune

Fast, portable, and opinionated build system



Opam

A source-based OCaml package manager



OCaml-LSP

LSP Server for OCaml



OCamlFormat

Auto-Formatter for OCaml Code



odoc

Documentation Generator for OCaml





Ppxlib

Standard library for PPX rewriters



Merlin

Editor helper, provides completion, typing, and source browsing in Vim and Emacs



Dune

Fast, portable, and opinionated build system



Opam

A source-based OCaml package manager



OCaml-LSP

LSP Server for OCaml



OCamlFormat

Auto-Formatter for OCaml Code



odoc

Documentation Generator for OCaml





Standard library for PPX rewriters



Editor helper, provides completion, typing, and source browsing in Vim and Emacs



Fast, portable, and opinionated build system



A source-based OCaml package manager



LSP Server for OCaml



Auto-Formatter for OCaml Code



Documentation Generator for OCaml





Standard library for PPX rewriters



Editor helper, provides completion, typing, and source browsing in Vim and Emacs



Fast, portable, and opinionated build system



A source-based OCaml package manager



LSP Server for OCaml



Auto-Formatter for OCaml Code



Documentation Generator for OCaml





Ppxlib

Standard library for PPX rewriters



Merlin

Editor helper, provides completion, typing, and source browsing in Vim and Emacs



Dune

Fast, portable, and opinionated build system



Opam

A source-based OCaml package manager



OCaml-LSP

LSP Server for OCaml



OCamlFormat

Auto-Formatter for OCaml Code



odoc

Documentation Generator for OCaml





Ppxlib

Standard library for PPX rewriters



Merlin

Editor helper, provides completion, typing, and source browsing in Vim and Emacs



Dune

Fast, portable, and opinionated build system



Opam

A source-based OCaml package manager



OCaml-LSP

LSP Server for OCaml



OCamlFormat

Auto-Formatter for OCaml Code



odoc

Documentation Generator for OCaml



- refmt
- dune-release

Melange is "simple"

Simplicity doesn't exclude
power

Simplicity doesn't exclude power

We need tools that do their job, but allow us to do ours

```
(melange.emit  
  (target artifact-directory))
```

```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```



```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```

```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```

```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```

```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```

```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```

```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```

```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```

```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocess melange.ppx reason-react-ppx)
  (compile_flags :standard --mel-cross-module-opt)
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js})
    (source_tree ./public)))
```



```
(melange.emit
  (target artifact-directory)
  (promote (until-clean))
  (alias my-js) ;; dune build @my-js
  (libraries reason-react)
  (preprocessor (standard met-cross-module-opt))
  (module_systems (es6 .mjs))
  (runtime_deps
    ./styles.css
    (glob_files_rec ./resources/*.{png,js}))
  (source_tree ./public)))
```

Simple, yet powerful

Simplicity promotes

- modularity
- composition

Simplicity promotes

- modularity
- composition
- **robust, reliable software**

"Simplicity is prerequisite
for reliability."

Edsger W. Dijkstra

Ahrefs is now built with Melange



Javier Chávarri · [Follow](#)

Published in Ahrefs · 7 min read · May 3, 2023



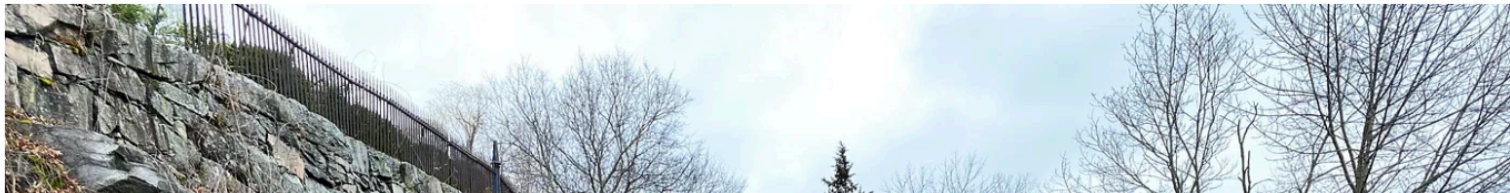
126



1



OCaml, all the way down



Collaboration drives excellence



Melange team (Portugal, circa August 2023)



Users make Melange better

New features must be driven by real-world use cases



↑ Share

☰ Examples

🔗 GitHub

📦 OPAM

Melange 4.0.0-52
OCaml 5.2.0
Reason 3.12.0
ReasonReact 0.15.0

OCaml Reason

☰ Format

⚡ Live </> JavaScript output

Loading...

This div has the ID selector `preview` .

Choose "React Greetings" in the Examples menu to see it in action, or
override by rendering into the element with
`ReactDOM.querySelector("#preview")`

Problems

No problems!



Console 0



dynamic import #1164



Open

anmonteiro wants to merge 9 commits into `main` from `anmonteiro/dynamic-import` 



Conversation 0

 Commits 9



Checks 5



Files changed 45



anmonteiro commented 3 weeks ago • edited ▼

fixes [#762](#)

↑ Share

☰ Examples

🐙 GitHub

📦 OPAM

Melange **4.0.0-52**
OCaml **5.2.0**
Reason **3.12.0**
ReasonReact **0.15.0**



 OCaml  Reason

☰ Format

 Live  JavaScript output

Loading...

Loading...

Problems



No problems!

Console 0



↑ Share

☰ Examples

🐙 GitHub

📦 OPAM

Melange 4.0.0-52
OCaml 5.2.0
Reason 3.12.0
ReasonReact 0.15.0



OCaml Reason

☰ Format

⚡ Live </> JavaScript output

Loading...

Loading...

Problems



No problems!

Console 0



Filters ▼

🔍 is:issue sort:reactions--+1-desc

✕ Clear current search query, filters, and sorts

🕒 222 Open ✓ 2,344 Closed

Author ▼

Label ▼

Assignee ▼

Sort ▼



Sourcemap support?

💬 32 👍 71

#1699 by jaredly was closed on Apr 24, 2020

Original code

0: z.ml



Generated code



```
1 let add_2 a b = a + b
```

```
1 // Generated by Melange
2 'use strict';
3
4
5 function add_2(a, b) {
6   return a + b | 0;
7 }
8
9 exports.add_2 = add_2;
10 /* No side effect */
11
```

What's my principle?

1. Build simple tools

That I want to use

1. Build simple tools

That I want to use

2. Invert from user needs

1. Build simple tools

That I want to use

2. Invert from user needs

3. Work with cool, talented people



x @_anmonteiro