

Introduction to ReasonML

Ryan Watkins

July 31, 2018

Javascript so far

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)

Problems

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)
- ▶ Template literals

Problems

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)
- ▶ Template literals
- ▶ Destructuring (single-branch pattern-matching)

Problems

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)
- ▶ Template literals
- ▶ Destructuring (single-branch pattern-matching)
- ▶ Promises + (async/await)

Problems

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)
- ▶ Template literals
- ▶ Destructuring (single-branch pattern-matching)
- ▶ Promises + (async/await)
- ▶ Spread operator

Problems

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)
- ▶ Template literals
- ▶ Destructuring (single-branch pattern-matching)
- ▶ Promises + (async/await)
- ▶ Spread operator
- ▶ Get/set on class definitions (some OOP)

Problems

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)
- ▶ Template literals
- ▶ Destructuring (single-branch pattern-matching)
- ▶ Promises + (async/await)
- ▶ Spread operator
- ▶ Get/set on class definitions (some OOP)

Problems

- ▶ No great pattern matching

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)
- ▶ Template literals
- ▶ Destructuring (single-branch pattern-matching)
- ▶ Promises + (async/await)
- ▶ Spread operator
- ▶ Get/set on class definitions (some OOP)

Problems

- ▶ No great pattern matching
- ▶ Non statically typed, i.e. you get typed errors at runtime

Javascript so far

Features

- ▶ ES6 (Babel + Webpack)
- ▶ Template literals
- ▶ Destructuring (single-branch pattern-matching)
- ▶ Promises + (async/await)
- ▶ Spread operator
- ▶ Get/set on class definitions (some OOP)

Problems

- ▶ No great pattern matching
- ▶ Non statically typed, i.e. you get typed errors at runtime
- ▶ Module system not great, you have to take care of it yourself

What is ReasonML?

What is ReasonML?

- ▶ It's Ocaml with JS-like syntax

What is ReasonML?

- ▶ It's Ocaml with JS-like syntax
- ▶ Contains pattern-matching

What is ReasonML?

- ▶ It's Ocaml with JS-like syntax
- ▶ Contains pattern-matching
- ▶ Several FP features (currying by default)

What is ReasonML?

- ▶ It's Ocaml with JS-like syntax
- ▶ Contains pattern-matching
- ▶ Several FP features (currying by default)
- ▶ Solid type system (30+ years of research)

What is ReasonML?

- ▶ It's Ocaml with JS-like syntax
- ▶ Contains pattern-matching
- ▶ Several FP features (currying by default)
- ▶ Solid type system (30+ years of research)
- ▶ Extremely fast compilation to JS (10x faster than TypeScript)

Ocaml to JS?

Ocaml to JS?

- ▶ Made possible by bucklescript

Ocaml to JS?

- ▶ Made possible by bucklescript
- ▶ Human readable output

Ocaml to JS?

- ▶ Made possible by bucklescript
- ▶ Human readable output
- ▶ Ocaml to Native and JS

Ocaml to JS?

- ▶ Made possible by bucklescript
- ▶ Human readable output
- ▶ Ocaml to Native and JS
- ▶ But we are not writing Ocaml, ReasonML is a javascript-like syntax of Ocaml.

Reason Primitives

Reason Primitives

Primitive	Example
Strings	'test'
Characters	'c'
Integers	23, -23
Floats	23.0, -23.0

Pattern Matching

Pattern Matching

Javascript already has single-branch pattern matching

I.e. `const foo, bar, baz = props`

Comparing Reason/Bucklescript to Elm

Comparing Reason/Bucklescript to Elm

- ▶ From the FAQ (paraphrased)

Comparing Reason/Bucklescript to Elm

- ▶ From the FAQ (paraphrased)
- ▶ TL;DR. Elm is pure, while Reason is pragmatic (support for React).

Comparing Reason/Bucklescript to Elm

- ▶ From the FAQ (paraphrased)
- ▶ TL;DR. Elm is pure, while Reason is pragmatic (support for React).

References / more useful information

References / more useful information