introduction to elm

content

an elm story

elm language + examples

integration

doc

q&a

an elm story

history

Elm is a strong **typed**, **functional** language for creating web-browser apps. It was created which usability, inmutability, performance, robutness and maintainability in mind. Everything began as a Master Thesis thanks to Evan Czaplicki.

who is using elm?

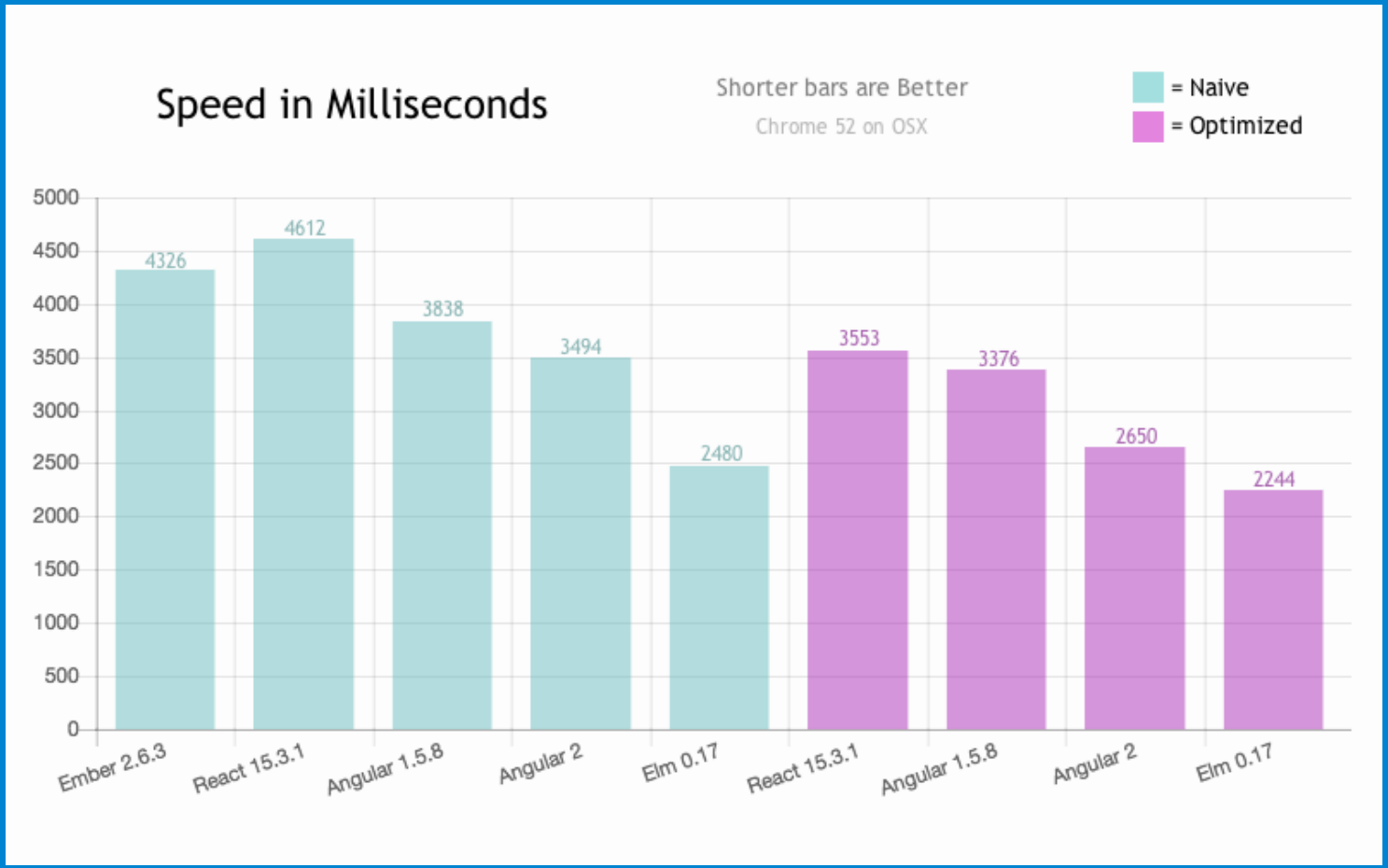
No runtime exceptions

“NoRedInk has 36k lines of Elm, and after more than a year in production, it still has not produced a single runtime exception.”

Elm online documentation

http://elm-lang.org/

Benchmarks


Speed in Milliseconds
Shorter bars are Better
Chrome 52 on OSX
= Naive
= Optimized
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
4326
4612
3838
3494
2480
3553
3376
2650
2244
Ember 2.6.3
React 15.3.1
Angular 1.5.8
Angular 2
Elm 0.17
React 15.3.1
Angular 1.5.8
Angular 2
Elm 0.17

Bye bye HTML, CSS and especially JS !! *

elm language

types

TYPE	EXAMPLE
String	“Hola”
Integer	1
Float	1.0
List	[1, 2, 3, 4, 5, 6]
Tuple	(1, 2.0, “tres”)
Record	{uno = 1, dos = 2.0, tres = “tres”}
Function	isOdd n = (n % 2) == 1

type aliases

```
type aliases My3DPoint =
  { x : Float
  , y : Float
  , z : Float
  , timestamp : Int
  , comment : String
  }
```

union types

```
type SimpleMsg = Increment | Decrement

type ComplexMessage = Increment Int | Decrement Int
```

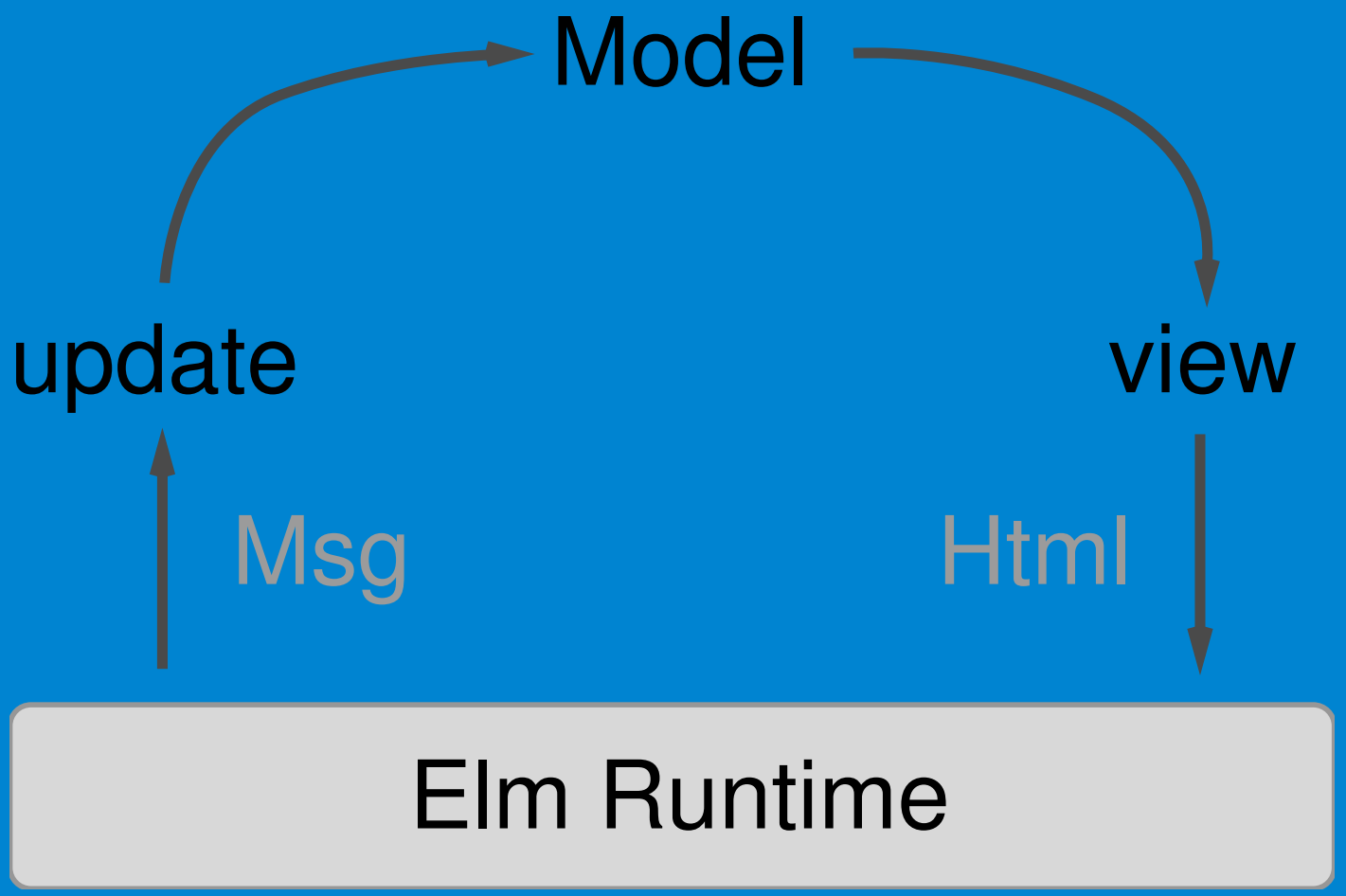
hello world

```
import Html exposing (text)

main =
  text "Hello, World!"
```

ecosystem

RESOURCE	WHAT IS IT?	EXAMPLE
elm-repl	REPL or Elm console	$ elm-repl
elm-reactor	Development compiler and server	$ elm-reactor –address=0.0.0.0
elm-make	Elm compiler	$ elm Main.elm –output=main.html
elm-package	Elm package manager	$ elm-package install elm-lang/http
Online editor	Elm online editor	http://elm-lang.org/try

elm architecture



model

```
-- It is a comment
model : Int
model =
  10
```

view

```
import Html exposing (div, button, text)
import Html.Events exposing (onClick)

-- Defining the message (only once)
type Msg = Increment | Decrement

view model =
div []
  [ button [ onClick Decrement ] [ text "-" ]
  , div [] [ text (toString model) ]
  , button [ onClick Increment ] [ text "+" ]
  ]
```

update

```
-- Defining the message (only once)
type Msg = Increment | Decrement

update msg model =
case msg of
  Increment ->   model + 1
  Decrement ->  model - 1
```

run time

```
import Html exposing (beginnerProgram)

main =
  beginnerProgram { model = model, view = view, update = update }
```

example

```
import Html exposing (beginnerProgram, div, button, text)
import Html.Events exposing (onClick)

main =
  beginnerProgram { model = model, view = view, update = update }

model =
  10

view model =
  div []
    [ button [ onClick Increment ] [ text "+" ]
    , div [] [ text (toString model) ]
    , button [ onClick Decrement ] [ text "-" ]
    ]

type Msg = Increment | Decrement

update msg model =
  case msg of
    Increment ->  model + 1
    Decrement ->  model - 1
```

CSS

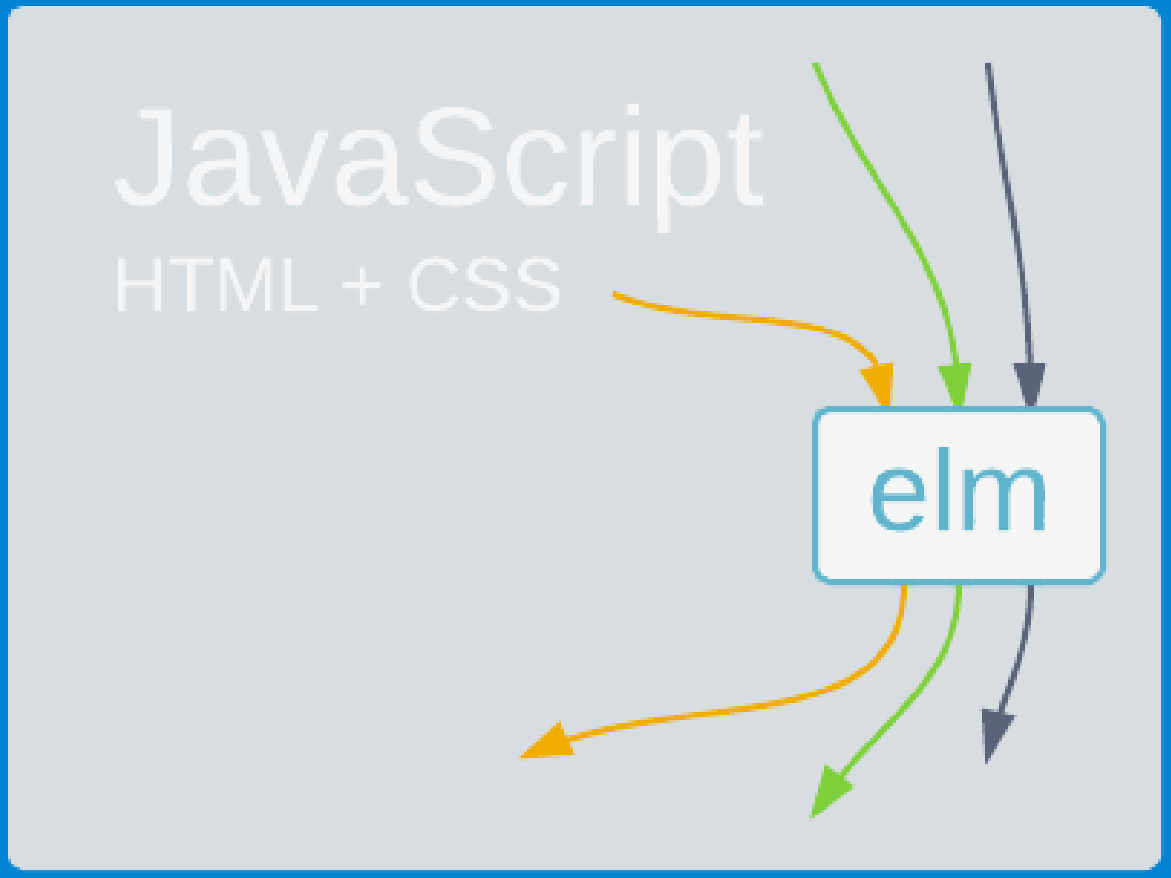
```
import Html exposing (beginnerProgram, div, button, text)
import Html.Events exposing (onClick)
import Html.Attributes exposing (style)

main =
  beginnerProgram { model = model, view = view, update = update }

model =
  10

view model =
  div [style [("backgroundColor", "blue"), ("text-align", "center"), ("width", "10%")]]
    [ button [ onClick Increment ] [ text "+" ]
    , div [] [ text (toString model) ]
    , button [ onClick Decrement ] [ text "-" ]
    ]

type Msg = Increment | Decrement

update msg model =
  case msg of
    Increment ->  model + 1
    Decrement ->  model - 1
```

complex messages

```
import Html exposing (beginnerProgram, div, button, text)
import Html.Events exposing (onClick)
import Html.Attributes exposing (style)

main =
  beginnerProgram { model = model, view = view, update = update }

diff = 10
model =  diff

view model =
  div [style [("backgroundColor", "blue"), ("text-align", "center"), ("width", "10%")]]
    [ button [ onClick (Increment diff)] [ text "+" ]
    , div [] [ text (toString model) ]
    , button [ onClick (Decrement diff) ] [ text "-" ]
    ]

type Msg = Increment Int | Decrement Int

update msg model =
  case msg of
    Increment value ->  model + value
    Decrement value ->  model - value
```

integration

html, css and js ↔ elm



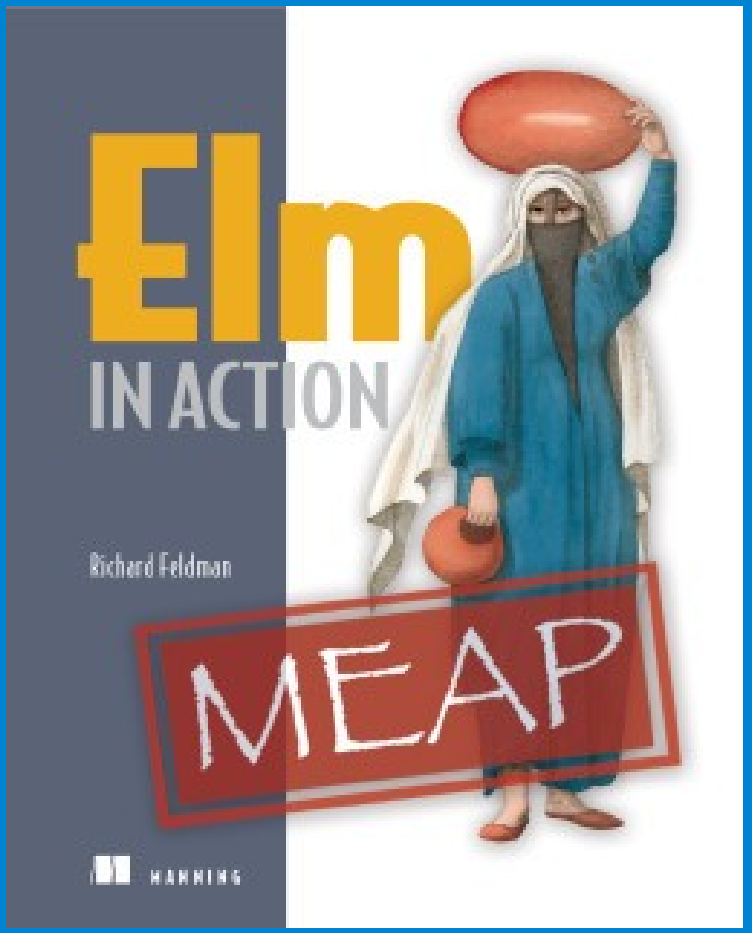
js communication

doc

web

- http://elm-lang.org
- http://tech.noredink.com/
- http://elmlang.herokuapp.com/
- https://dennisreimann.de/articles/elm.html
- https://github.com/isRuslan/awesome-elm
- https://www.reddit.com/r/elm/
- https://www.elm-tutorial.org/

book



Q&A