Intro to Elm

London JS Community

HELLO!

My name is Mario

Yehuda Katz
@wycats

Following

Front end software development is:

- real-time (instant load, 60fps)
- distributed, incremental (synchronize remote data as needed)
- asynchronous
- reactive (react to user actions in realtime)

Front end is the hardest kind of dev I do. The folks who do it every day are heroes.

2:53 AM - 15 Nov 2017

1,670 Retweets **4,485** Likes

WEBASSEMBLY

Global 65.34%

IE	Edge *	Firefox	Chrome	Safari	iOS Safari *	Opera Mini *	Chrome Android	UC for Android	Samsung Internet
		4 52	49						
		55	60		10.2				
	3 15	56	61	10.1	10.3				4
11	16	57	62	11	11	all	62	11.4	5
	17	58	63	TP					
		59	64						
		60	65						

What might a browser language look like if we designed it from scratch?

A delightful language for reliable webapps.

Generate JavaScript with great performance and no runtime exceptions.

Javascript	Elm
npm / yarn	Built in
React / Preact / Vue	Built in
Redux	Built in
Immutable.js	Built in
Typescript/Flow	Built in

“Redux evolves the ideas of Flux, but avoids its complexity by taking cues from Elm.”

- **Dan Abramov, Author of Redux**

Source: http://redux.js.org/

Why not language X?

Elm as a language was designed specifically for webapps

The plan

now ← What is Elm?

then ← What is nice about Elm?

after ← How do I write Elm?

finally ← What if Elm is not for me?

Intro

What is Elm?

- A functional language
- Compiles to Javascript
- Statically typed + inferred
- All data is immutable
- All functions are pure
- ‘null’ / ‘undefined’ / Exceptions do not exist

What is nice about Elm?

Nice thing:

Beautiful, helpful, friendly errors

```
-- NAMING ERROR ---------------------------------------- teach/ErrorMisname.elm

Cannot find variable `List.nap`.

6|   List.nap identity (List.range 1 10)
     ^^^^^^^^
`List` does not expose `nap`. Maybe you want one of the following?

    List.map
    List.any
    List.map2
    List.map3

Detected errors in 1 module.
```

```
-- TYPE MISMATCH ------------------------------------- teach/ErrorMultiIf.elm

The 2nd and 3rd branches of this `if` produce different types of values.

 5|   if n < 0 then
 6|     "negative"
 7|   else if n > 0 then
 8|     "positive"
 9|   else
10|>    42

The 2nd branch has this type:

    String

But the 3rd is:

    number

Hint: All the branches of an `if` need to match so that no matter which one we
take, we get back the same type of value overall.

Detected errors in 1 module.
```

```
-- TYPE MISMATCH ------------------------------------ teach/ErrorTruthiness.elm

This condition does not evaluate to a boolean value, True or False.

9|     if String.length user.name then
           ^^^^^^^^^^^^^^^^^^^^^^^^^
You have given me a condition with this type:

    Int

But I need it to be:

    Bool

Hint: Elm does not have "truthiness" such that ints and strings and lists are
automatically converted to booleans. Do that conversion explicitly.

Detected errors in 1 module.
```

```
-- TYPE MISMATCH ---------------------------------------- teach/ErrorConcat.elm

The left argument of (+) is causing a type mismatch.

5|    "Name: " + repo.name
      ^^^^^^^^
(+) is expecting the left argument to be a:

    number

But the left argument is:

    String

Hint: To append strings in Elm, you need to use the (++) operator, not (+).
<http://package.elm-lang.org/packages/elm-lang/core/latest/Basics#++>

Detected errors in 1 module.
```

Nice thing:

No runtime exceptions in practice

rtfeldman

200,000 lines, and more like 2.5 years 😉

still no runtime exceptions!

I try to say "exceptions" and not "errors" because we still make business logic mistakes and such 😄

noredink

Sure! I'll get some numbers from you when I'm back at my desk. Spoiler: the number of runtime errors is zero.

Excluding dependencies and tests:
193 modules
19,247 significant lines of code

Tests:
66 modules
5,241 significant lines of code

Culture Amp

Me:

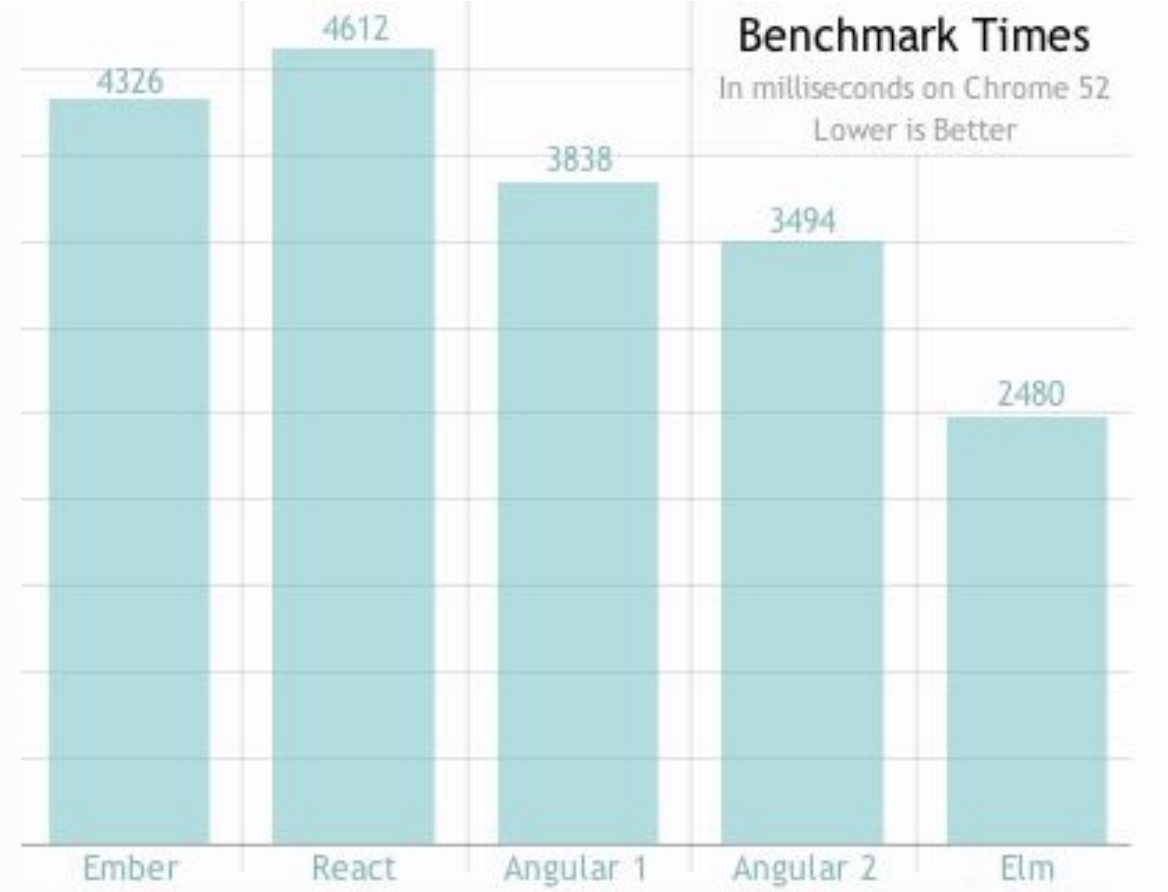
~10,000 LOC

~5 projects

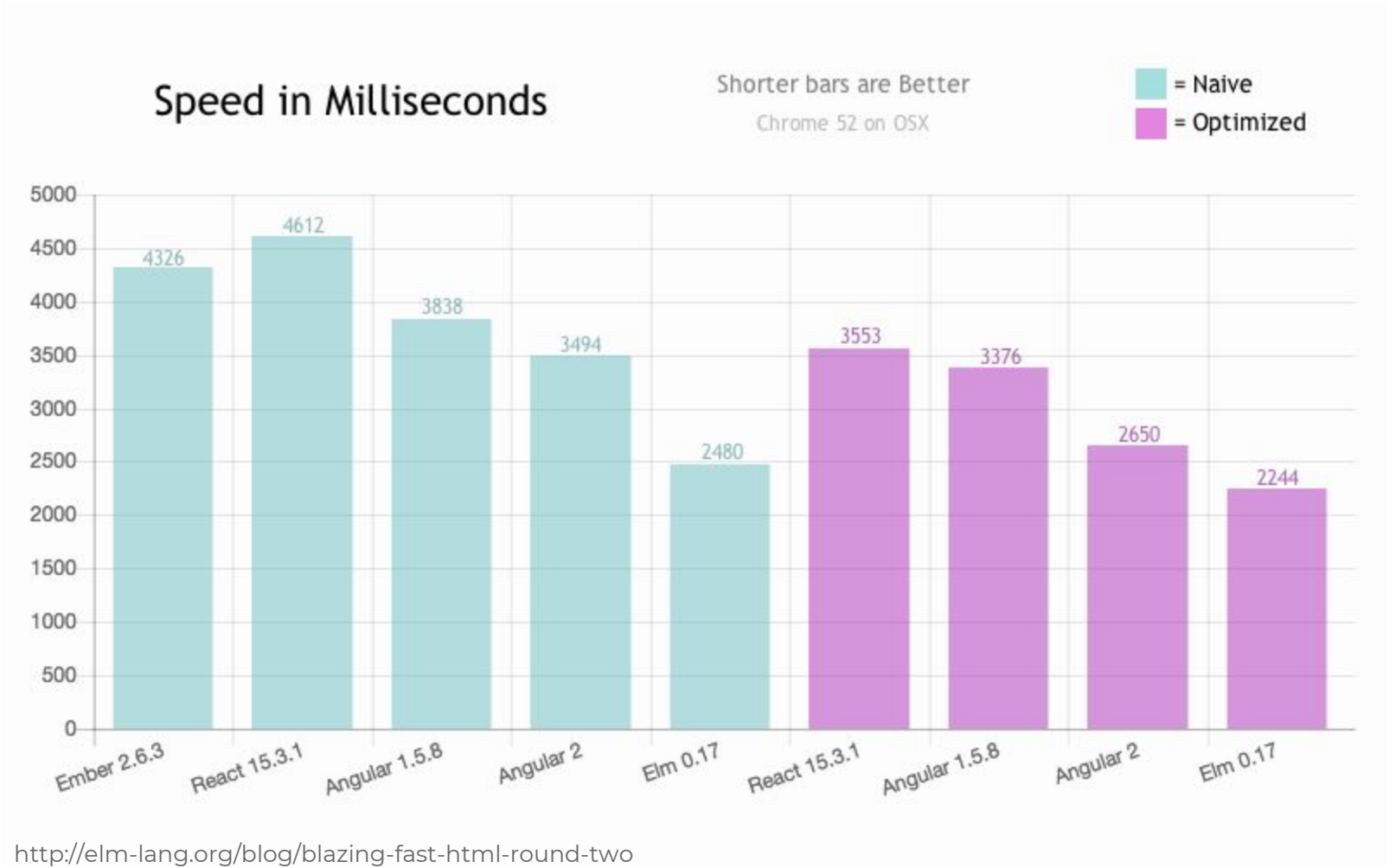
“If it compiles, it works.”

Nice thing:

Performance



http://elm-lang.org/blog/blazing-fast-html-round-two



http://elm-lang.org/blog/blazing-fast-html-round-two

Nice thing:

Automatic-semvar for Elm packages

elm packages

community

elm-community/array-extra

... 1.0.2 — Overview

Convenience functions for working with Array

elm-community/basics-extra

... 1.2.0 ... 2.1.2 — Overview

Additional basic functions

elm-community/dict-extra

... 1.5.0 ... 2.2.0 — Overview

Convenience functions for working with Dict

elm-community/easing-functions

... 1.0.2 — Overview

Easing functions for animations.

elm-community/elm-check

... 1.0.2 ... 2.0.1 — Overview

[deprecated] Property-based testing in Elm

elm-community/elm-datepicker

... 5.0.1 ... 6.1.0 ... 7.2.1 — Overview

Resources

- Fancy Search
- Using Packages
- API Design Guidelines
- Write great docs
- Preview your docs
- Elm Website

Popular Packages

General
- core
- http

Rendering
- html
- svg
- markdown

Effects
- dom
- navigation
- geolocation
- page-visibility
- websocket

User Input
- mouse
- window

```
$ elm package diff mdgriffith/style-elements 3.4.1 4.2.1
Comparing mdgriffith/style-elements 3.4.1 to 4.2.1...
This is a MAJOR change.

------ Added modules - MINOR ------

    Element.Input


------ Changes to module Element - MAJOR ------

    Added:
        type alias GridPosition style variation msg =
            { start : (Int, Int),
              width : Int,
              height : Int,
              content : Element.Element style variation msg
            }
        type alias NamedGridPosition =
            Style.Internal.Model.NamedGridPosition
        cell : Element.GridPosition style variation msg -> Element.OnGrid (Element.Element style variation msg)
        decorativeImage : style -> List (Element.Attribute variation msg) -> { src : String
                                                                              } -> Element.Element style variation msg
        download : String -> Element.Element style variation msg -> Element.Element style variation msg
        downloadAs : { src : String,
                       filename : String
                     } -> Element.Element style variation msg -> Element.Element style variation msg
```


1.4.2
Patch
Addition
Removal/change

Nice thing:

Wonderfully helpful & nice community


elm
Europe 2017
elmeurope.org
efrei
efrei

#general

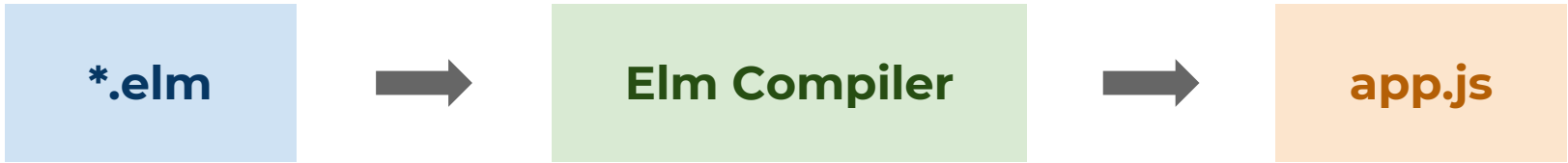
☆ | 11,233

https://elmlang.herokuapp.com

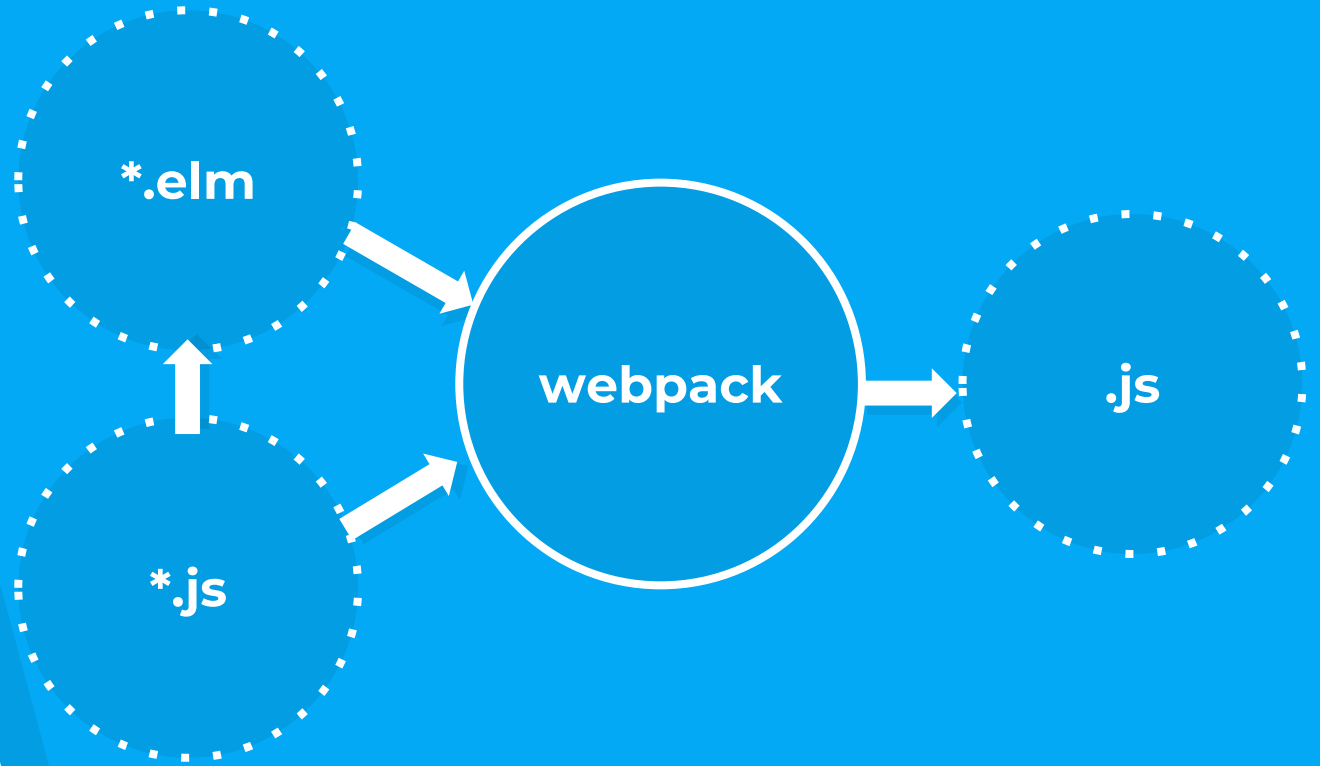
#beginners

@supermario / #london

How do I write Elm?


*.elm
Elm Compiler
app.js

Compiling



elm-webpack-loader

Webpack loader for the Elm programming language

With plain javascript

```
let Elm = require('./App.elm'),
    node = document.getElementById('main');

Elm.App.embed(node);
```

With react-elm-components

```
import Elm from 'react-elm-components'
import { Todo } from '../dist/elm/todomvc.js'

function render() {
  return <Elm src={Todo} />
}
```

OKAY

Let’s code!

Coding Time

Basic +/- counter

```elm
module Main exposing (..)

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


initModel = 0

type Msg = Increment | Decrement

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

    Decrement ->
      model - 1

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

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

We define Model Msg values

We write Update View functions

Elm does the rest!

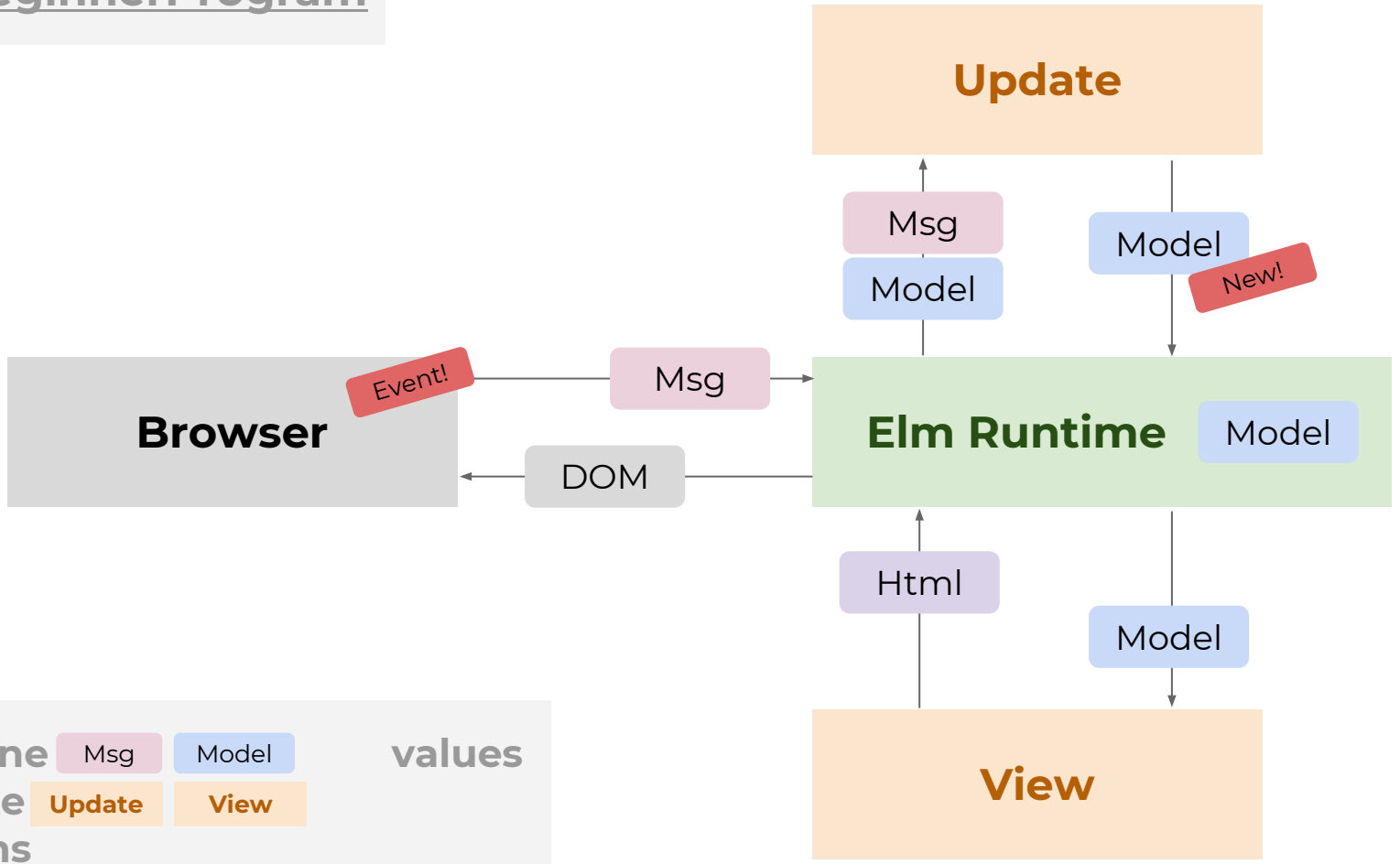
State

View

Update

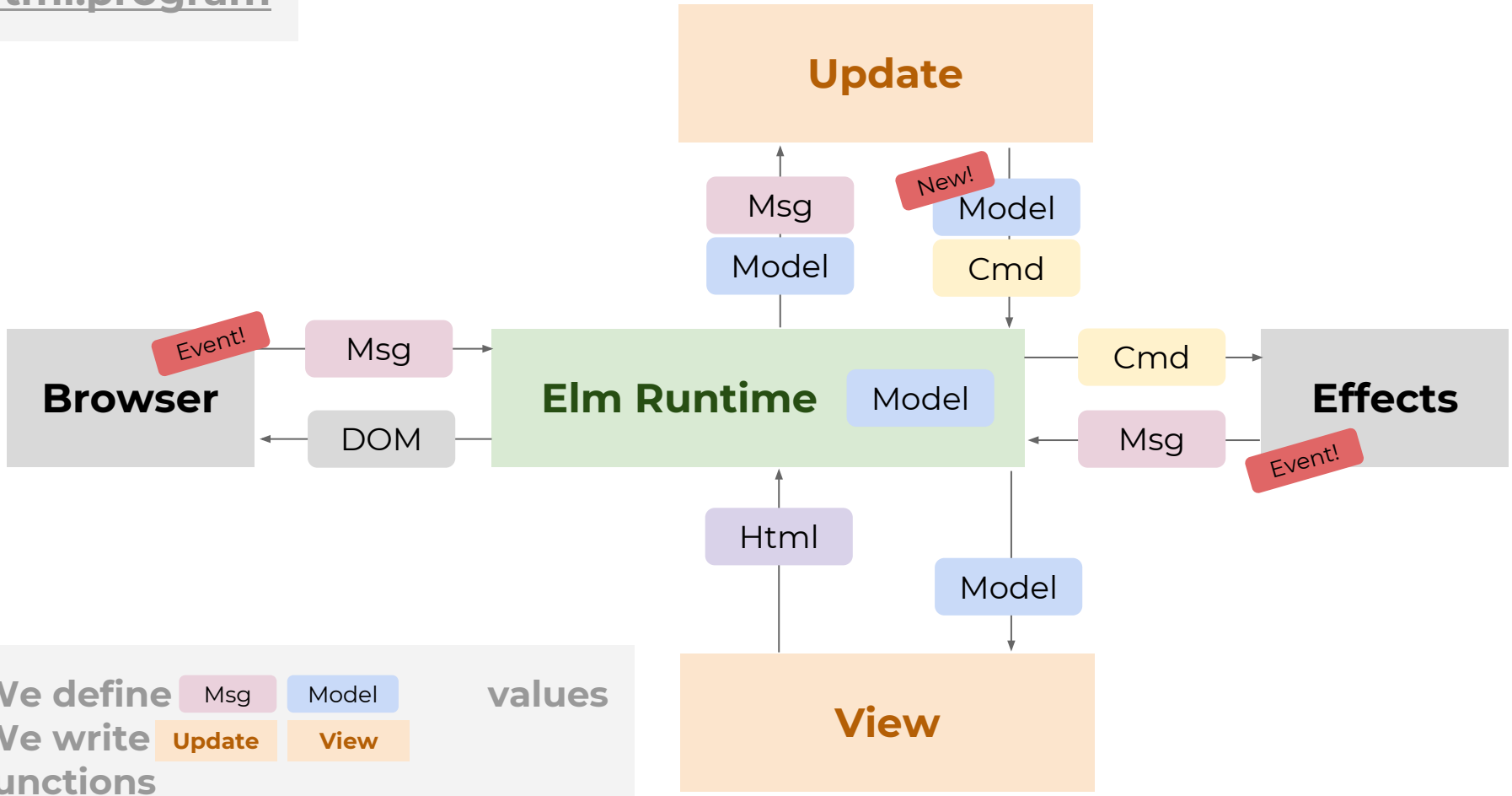
```
main = beginnerProgram { model = initModel , view = view , update = update }
```

Html.beginnerProgram



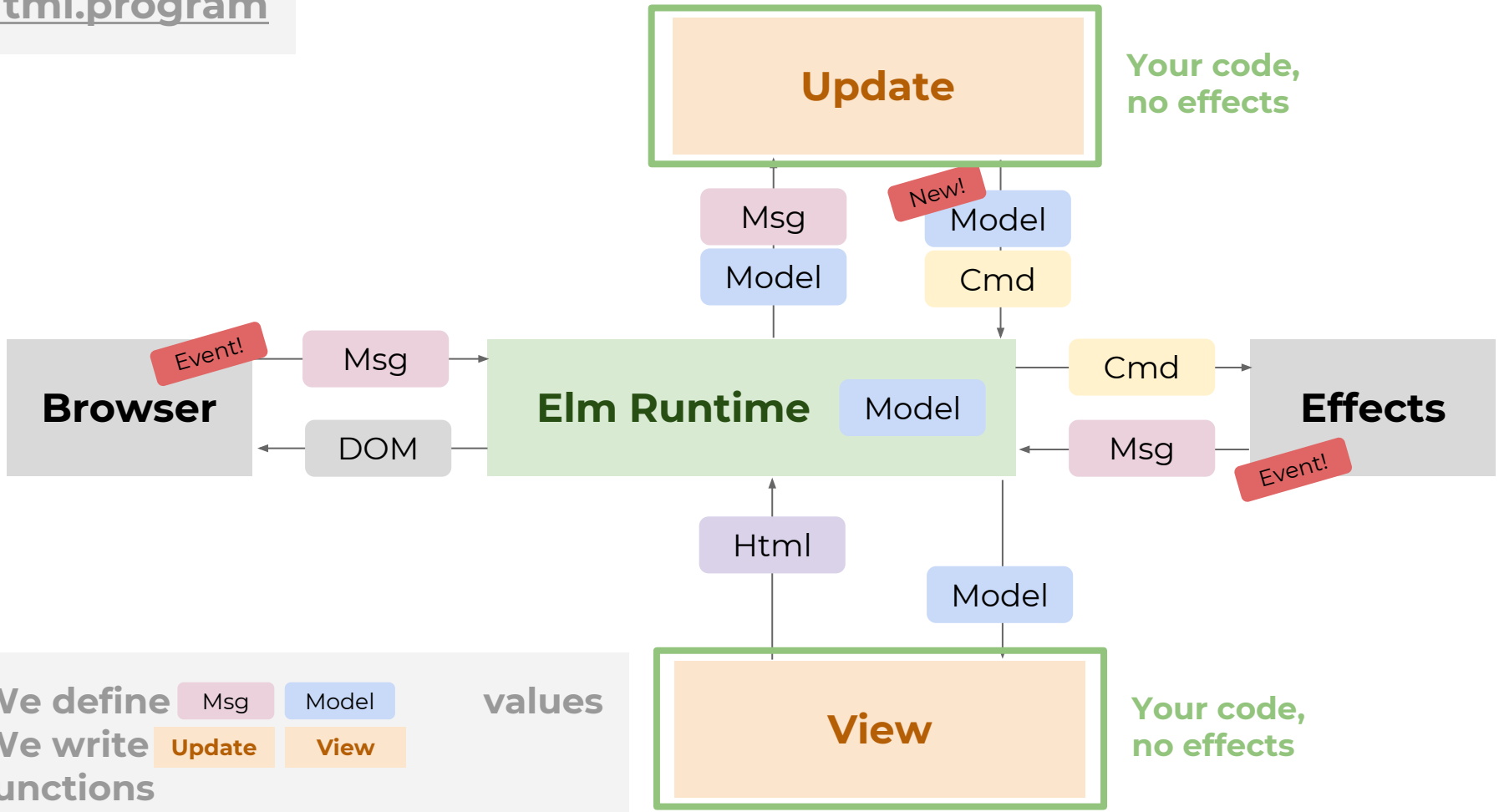
We define Msg Model values
We write Update View functions
Elm does the rest!

Html.program



We define Msg Model values
We write Update View functions
Elm does the rest!

Html.program



We define Msg Model values
We write Update View functions
Elm does the rest!

Nice thing:

Time-travelling debugger

Nice thing:

Type inference

Nice thing:

elm-format

What if Elm is not for me?

- “General Purpose Language”
- Fiddling with the DOM
- Javascript interop is strict

Recap

- What is Elm?
- What is nice about Elm?
- How do I write Elm?
- What if Elm is not for me?

Come learn Elm!

Elm-London-Meetup

- Learn with other beginners
- Get help with your projects
- Pair with advanced folk

Elm London
Code Night

Wed, 6 December
@Expedia

http://meetup.com/elm-London-Meetup

Questions?

Thanks!

@supermario / #london

@realmario

Elm-London-Meetup