



meetup

Elm London

Code Night

2017-10-04

HELLO!

@supermario on elmlang.slack.com



meetup

[Elm-London-Meetup](#)

A big thanks to

carwow

**For venue, food &
drinks!**

Agenda

now ← **Hello!**

then ← **Elm Intro / Lightning talks**

~7:30 ← **Food & social**

~8:00 ← **Hack!**

9:30 ← **Fin, Pub!**



Mike Wagg

Elm Conf US Report



James Porter

Data vis in Elm - Lessons Learnt



Let's eat!
(and socialise)



Let's code!



- **Beginners: Intro**
- **Intermediate: find people!**
- **Advanced: help people!**



Elm Crash-course

HELLO!

My name is Mario

I work at **Travelport** Locomote

github.com/supermario

Goals:

- **Things that might not be obvious**
- **Awesome things**
- **Pitfalls**
- **References for more**



Agenda:

- **What is Elm?**
- **Let's see some code!**
- **The Elm Architecture**
- **Awesome things**
- **How do I use Elm at work?**
- **Caveats**



Intro

What is Elm? 



elm

A delightful language for reliable webapps.

Generate JavaScript with great performance and no runtime exceptions.

- **A functional language**
- **Statically typed + inference**
- **All data is immutable**
- **All functions are pure**
- **'null' / 'undefined' / Exceptions do not exist**

Javascript

npm

Webpack

React

Redux

Typescript/Flow

Immutable.js

Elm

Built in

Built in

Built in

Built in

Built in

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/>

Side note:

**“Unbreakable” &
“Functional”**

**What do we mean
when we say **breakable**?**

“The sky is green.”

Wrong?

**“The sky is Uncaught
TypeError: Cannot read
property 'color' of null at
<anonymous>:1:5”**

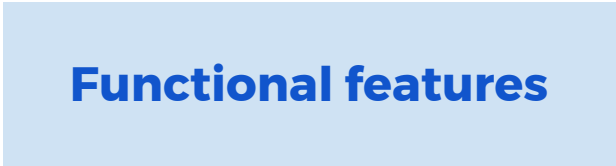
Broken.

Functional Style



Here FP practices require discipline

Functional Language



Here FP practices are the default

OKAY

Let's code!

Code Time

Basic +/- counter

```
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**

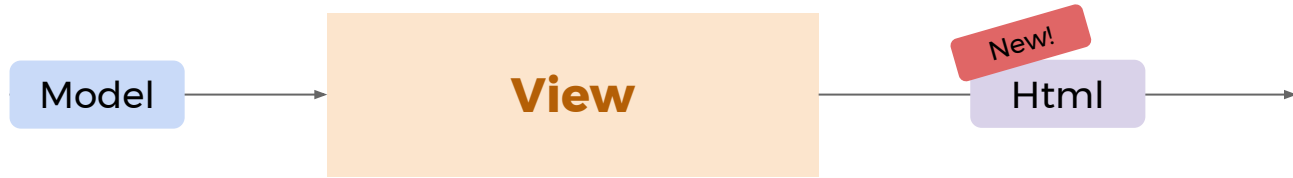
Elm does the rest!

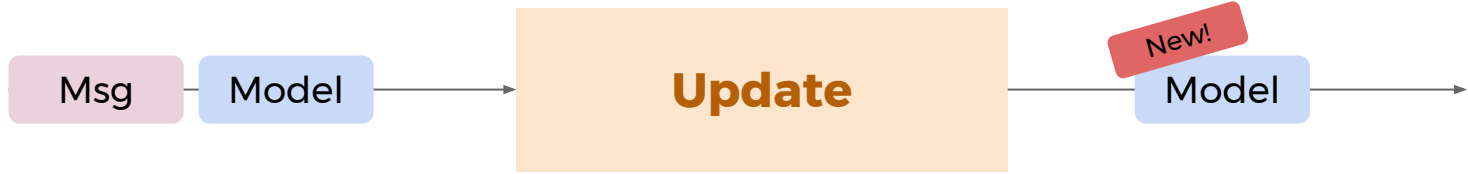
State

View

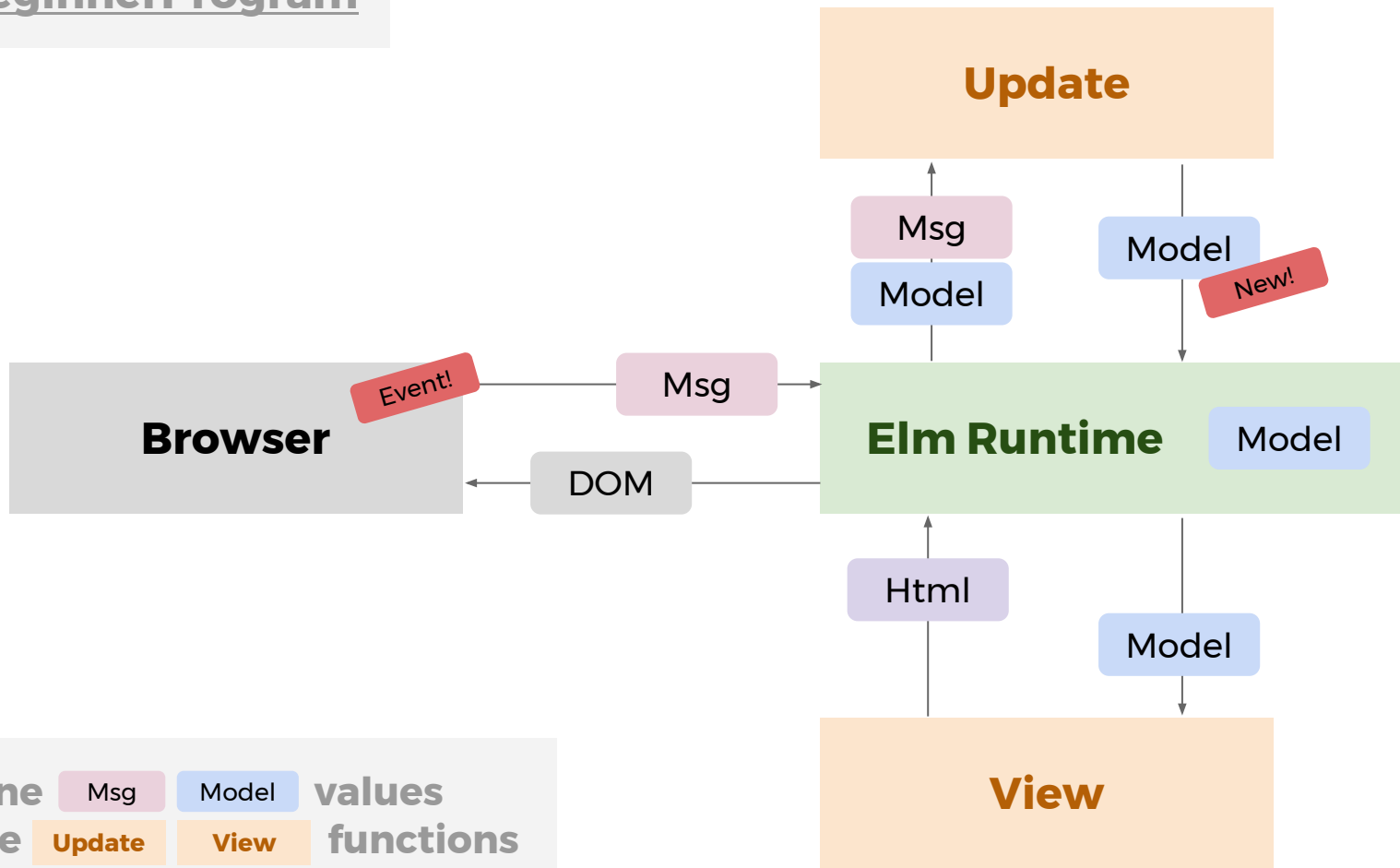
Update

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



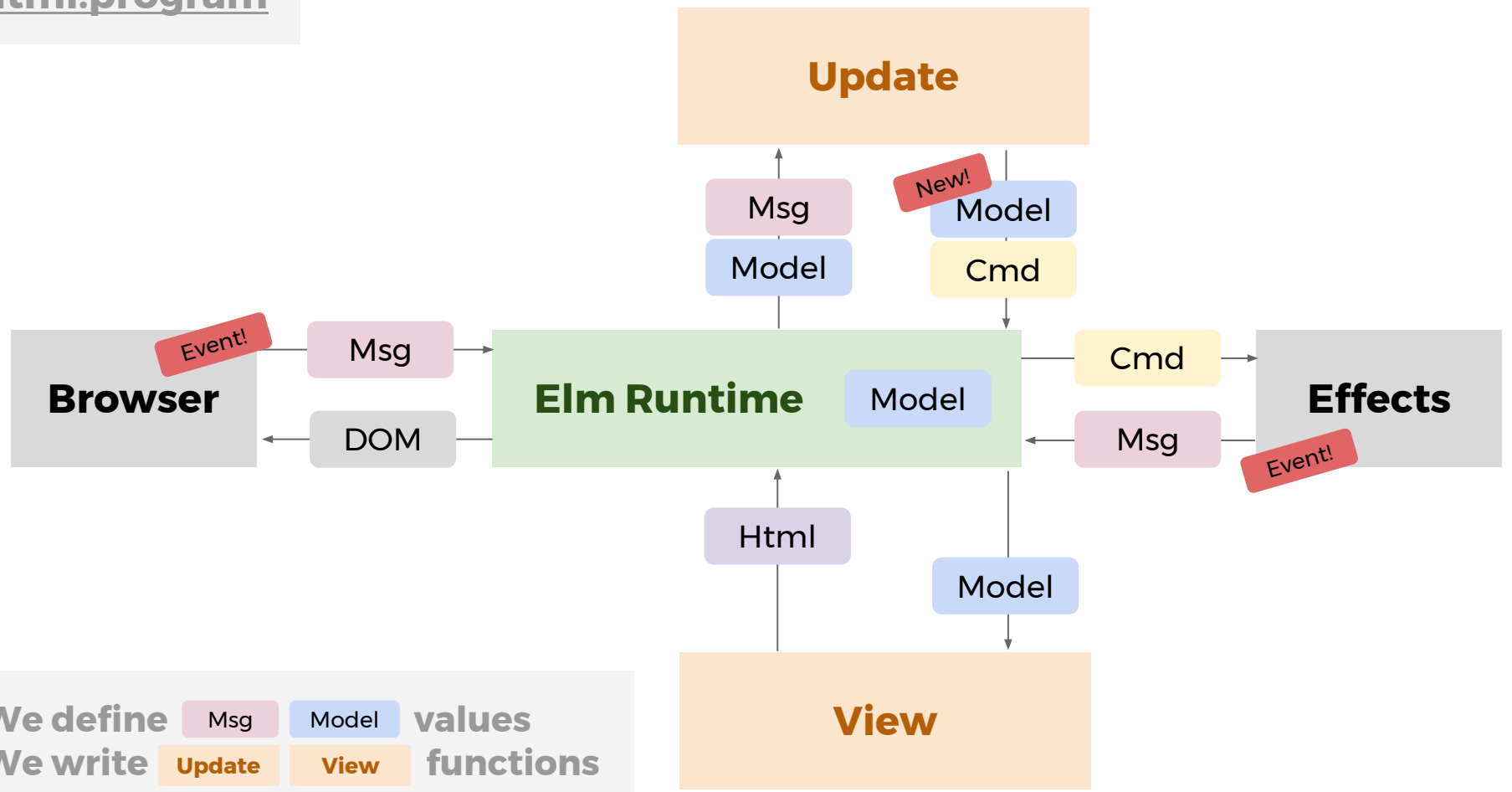


Html.beginnerProgram



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!



Next:

Awesome things!

Next:

**Best pair-programming
buddy ever**

Refactoring:

**Nice canned
scenarios**

Refactoring:

**Worst possible
scenario**

Next:

Elm packages
Auto-semvar

[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](#)

Next:

elm-format

Next:

Compiling Elm 

***.elm**

```
graph LR; A[*.elm] --> B[Elm Compiler]; B --> C[app.js]
```

A horizontal flowchart illustrating the compilation process. It consists of three rectangular boxes connected by right-pointing arrows. The first box is light blue and contains the text ***.elm**. A dark grey arrow points from this box to a light green box in the center containing the text **Elm Compiler**. A second dark grey arrow points from the green box to a light orange box on the right containing the text **app.js**.

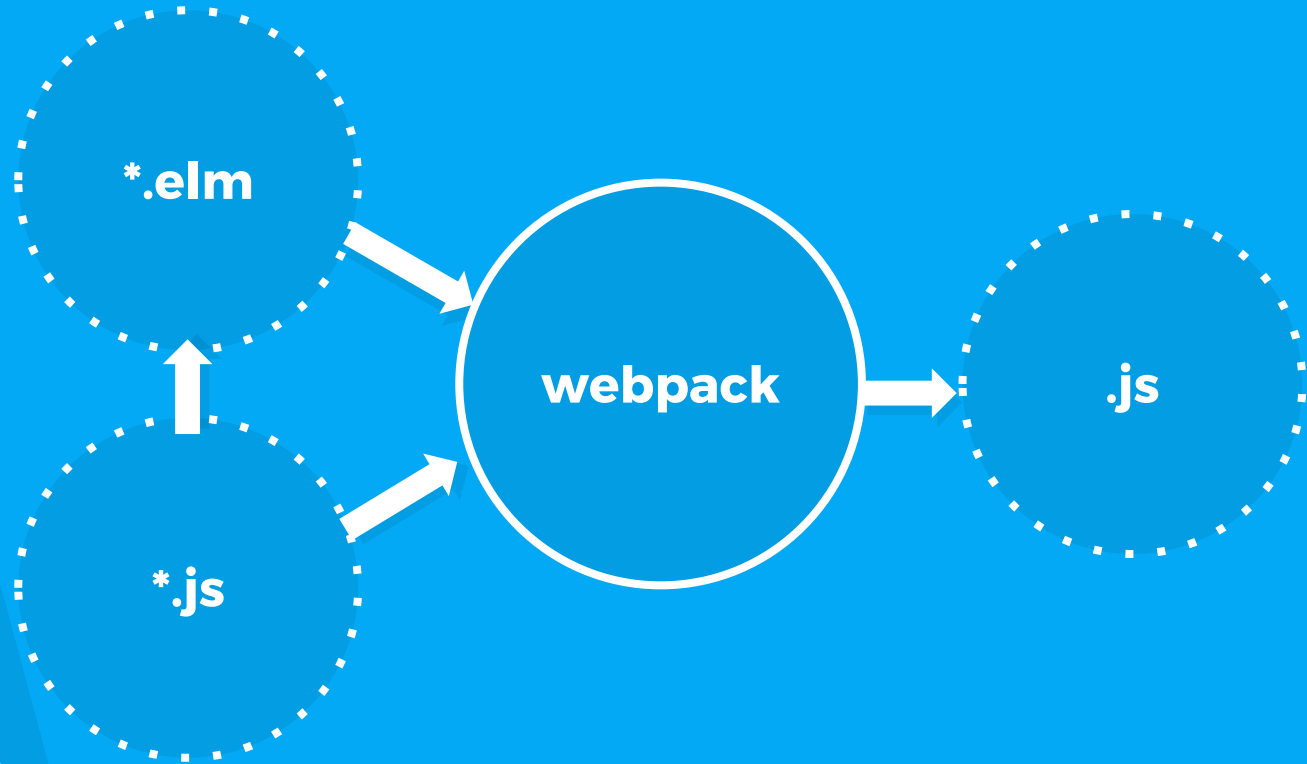


Elm Compiler



app.js

Compiling



elm-webpack-loader

Webpack loader for the Elm programming language

Next:

How do I use Elm  at work?

Answer:

Gradually!

<http://elm-lang.org/blog/how-to-use-elm-at-work>

With react-elm-components

```
import Elm from 'react-elm-components'  
import { Todo } from '../dist/elm/todomvc.js'  
  
function render() {  
  return <Elm src={Todo} />  
}
```

With plain javascript

```
var Elm = require('../dist/elm/todomvc.js');  
var app = Elm.Todo.embed(node);
```


Next:

Caveats

- **“General Purpose Language”**
- **Fiddling with the DOM**
- **Javascript interop is strict**
- **You may be more frustrated
with other languages**

Thoughts after 2 years of FP

- **Not a silver bullet - but trades off for better problems!**
- **Yet to see a broken runtime**
- **Elm reveals your mistakes**
- **Refactoring is awesome**

Further reading/watching

- The Elm Guide <https://guide.elm-lang.org/>
- [Developer Happiness on the Front End with Elm](#) - Kevin Yank
- [The life of a file](#) - Evan Czaplicki
- [Scaling Elm apps](#) - Richard Feldman

THANKS!



@**supermario** on elmlang.slack.com



meetup [Elm-London-Meetup](https://www.meetup.com/Elm-London-Meetup/)

github.com/supermario