

Building web apps with Elm

Visit README for demo and exercise URLs:
bit.ly/fluent-elm-18

Jeremy Fairbank

 @elpapapollo /  jfairbank

td

@testdouble helps improves
how the world build software.
testdouble.com/agency

The
Pragmatic
Programmers

Programming Elm

Build Safe and
Maintainable
Front-End
Applications

Jeremy Fairbank
edited by Brian MacDonald

bit.ly/programming-elm



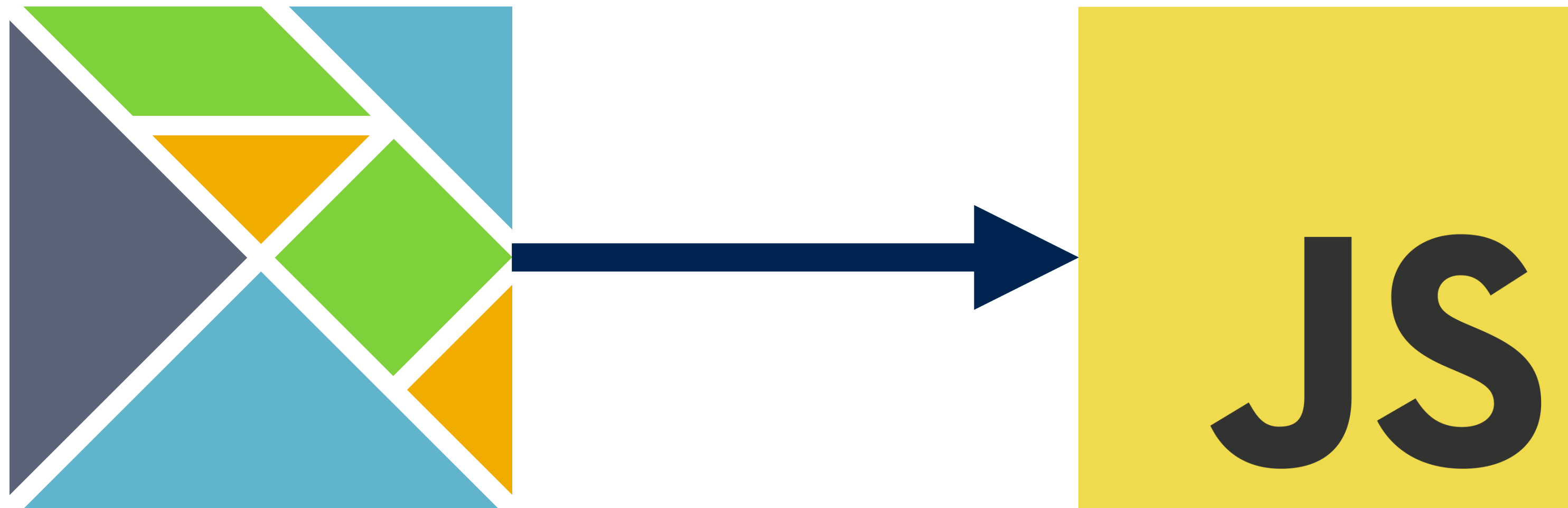
elm

*Functional and statically typed
programming language for frontend
development*

Web and UI Focused



Compiles to JavaScript





*No runtime exceptions in
practice.*

Compile time static type checks

The 2nd argument to function `add` is causing a mismatch.

```
7|      add 2 "3"  
      ^^^
```

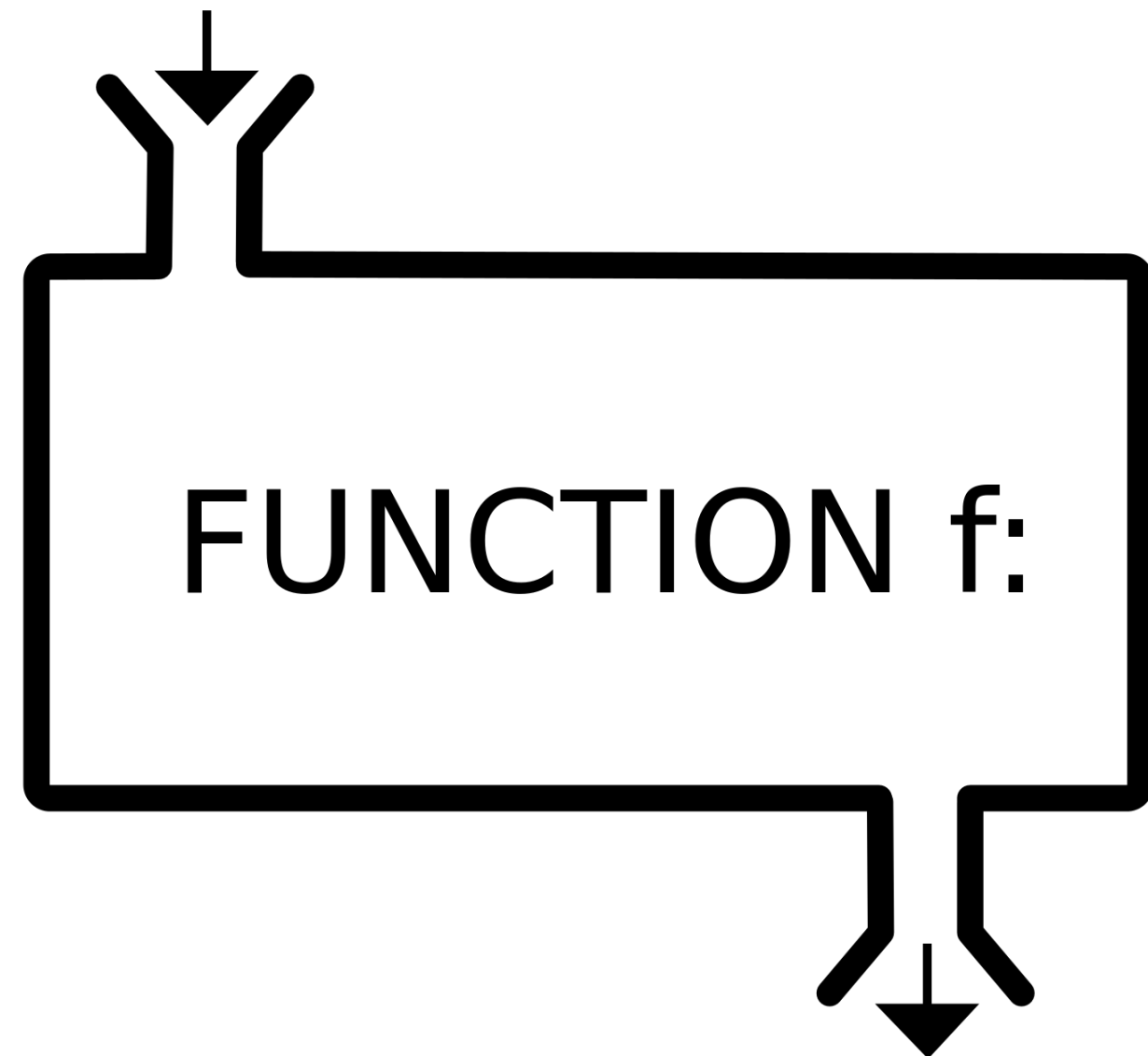
Function `add` is expecting the 2nd argument to be:

Int

But it is:

String

INPUT x



OUTPUT $f(x)$

Functional

Pure

Data in

Data out

Pure

*No side
effects*

Pure

*Predictable
and Testable!*



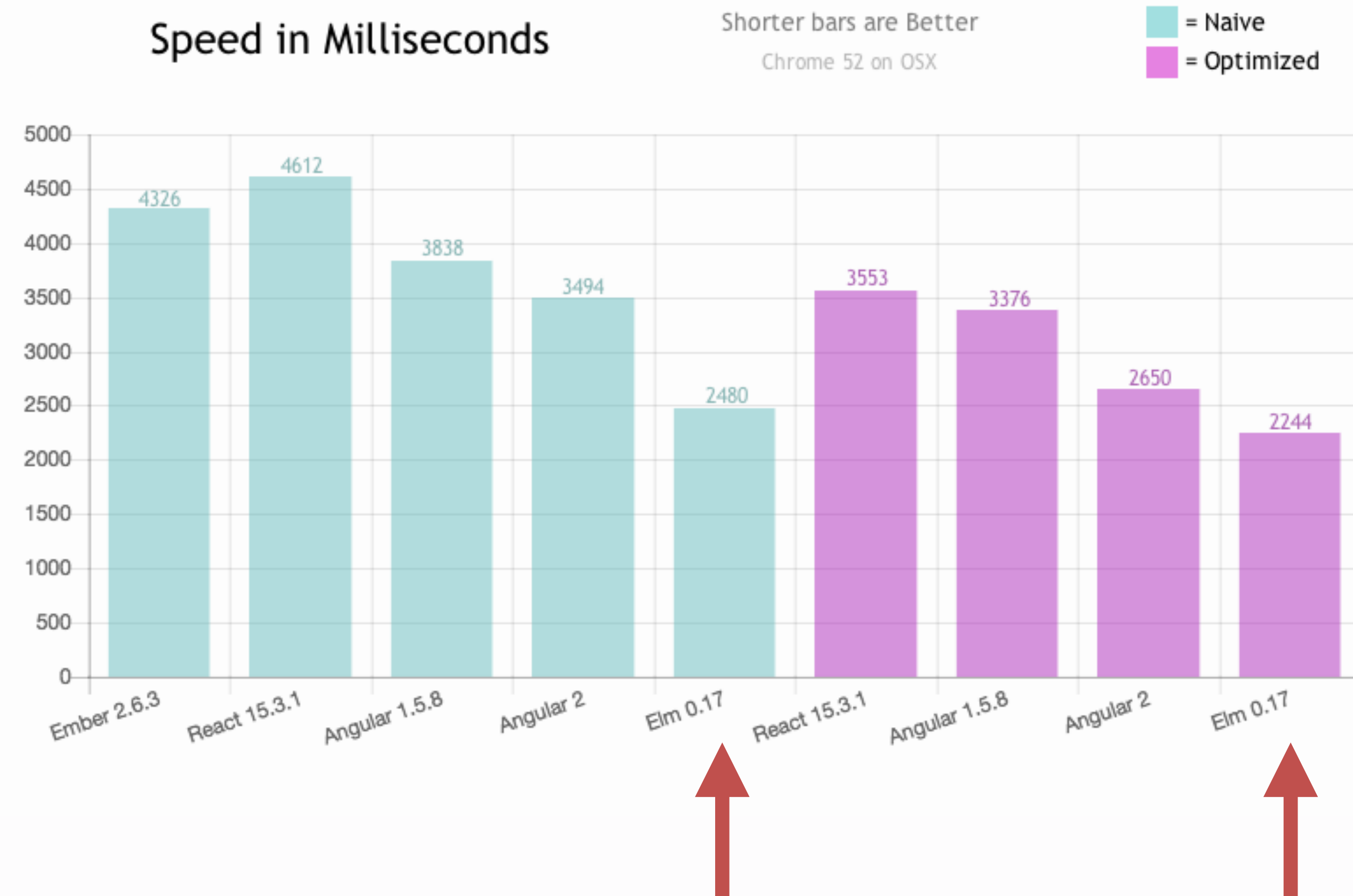
Immutable Data

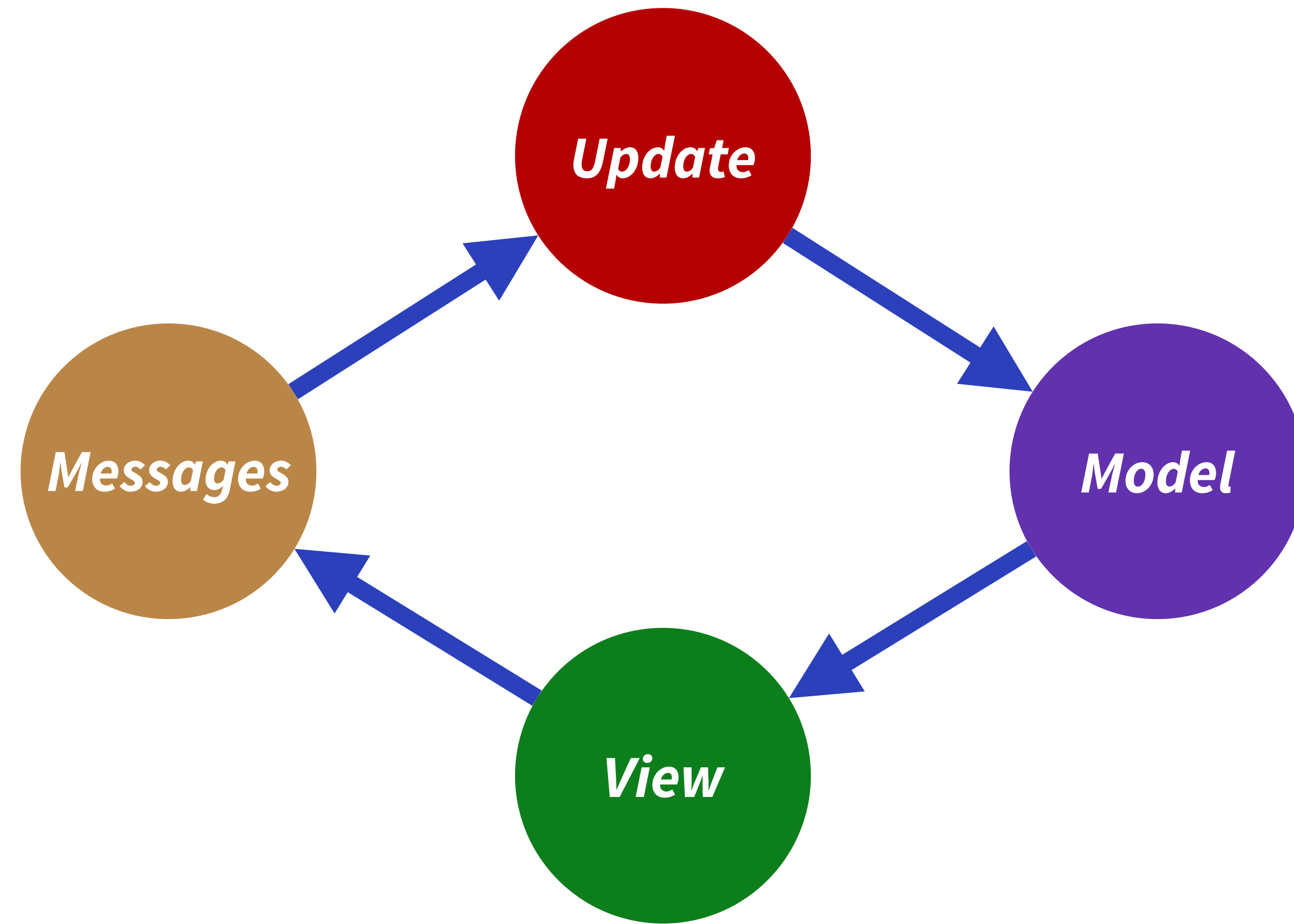
- Safety and consistency
- Explicit flow of data
- No subtle mutation bugs



No undefined is not a
function

Fast





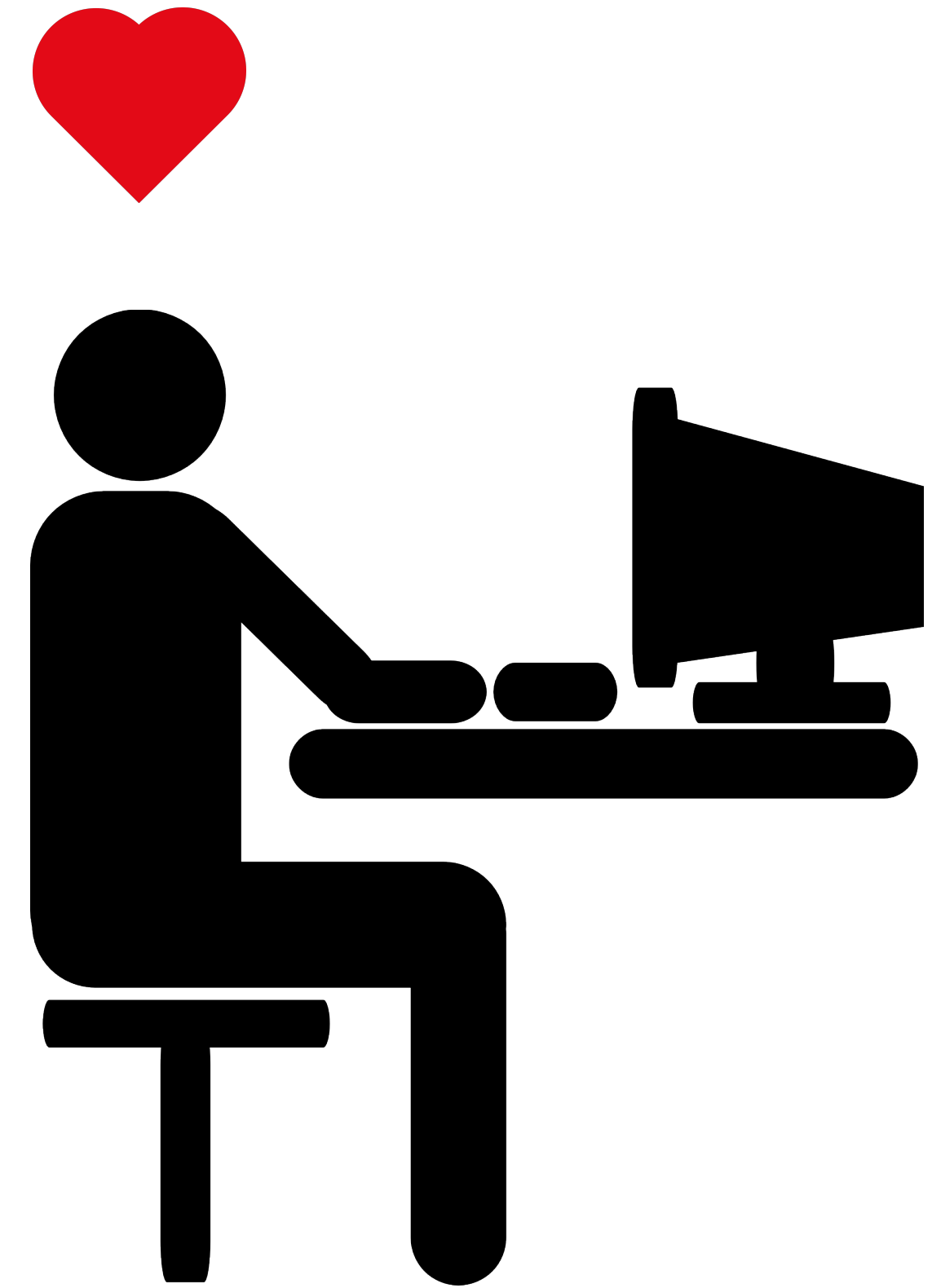
One framework.

No fatigue.

✓ *Easier to write code*

✓ *Easier to refactor*

✓ *Easier to write tests*



Content

- Functions and expressions
- Static applications
- Dynamic applications with the Elm Architecture
- Working with Lists
- Static Types and Type Aliases
- Input Events and Union Types

Format

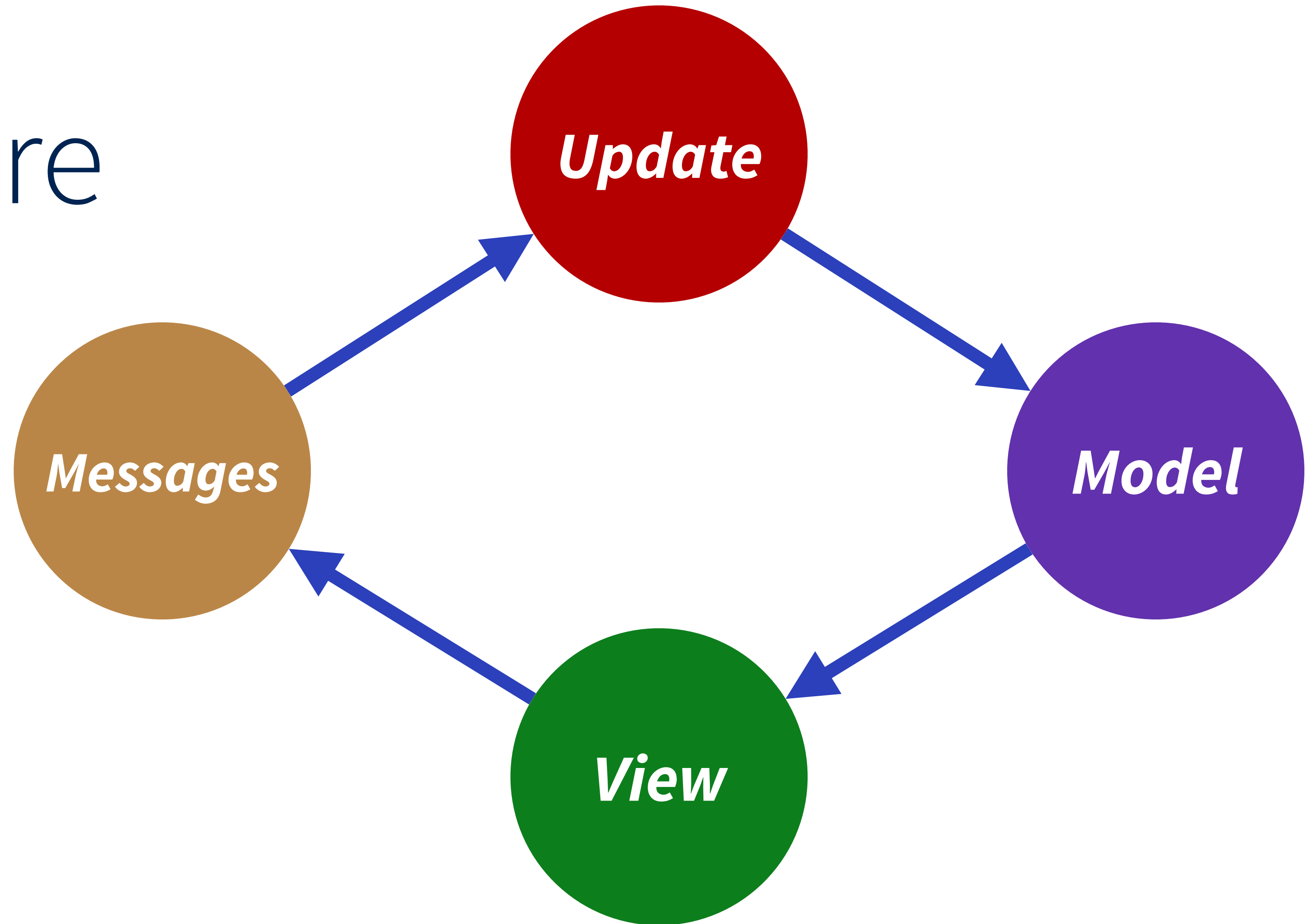
- Demos via ellie-app.com
- Exercises via ellie-app.com
- Q&A

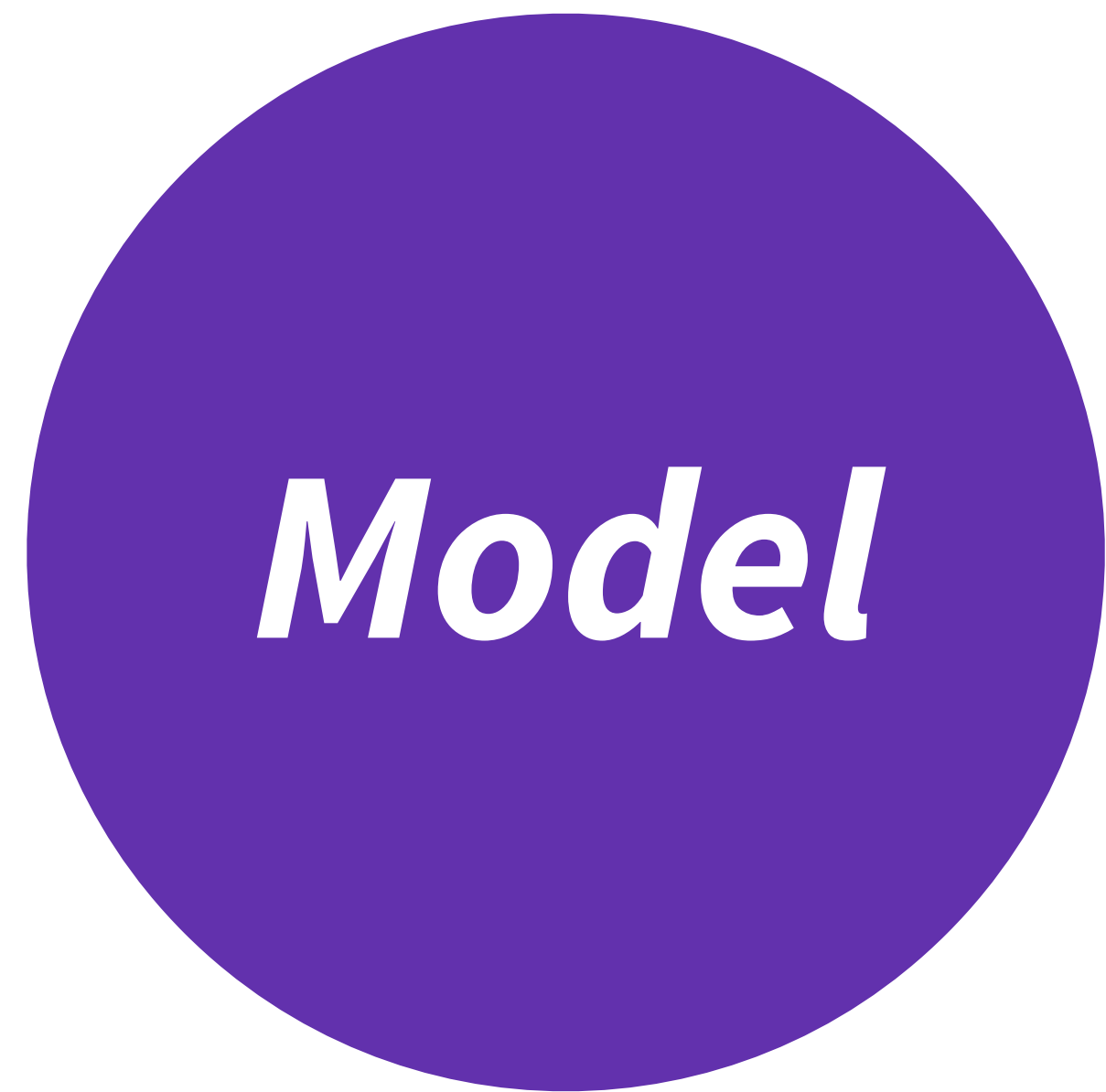
Coding Time!

README for demo and exercise URLs:

bit.ly/fluent-elm-18

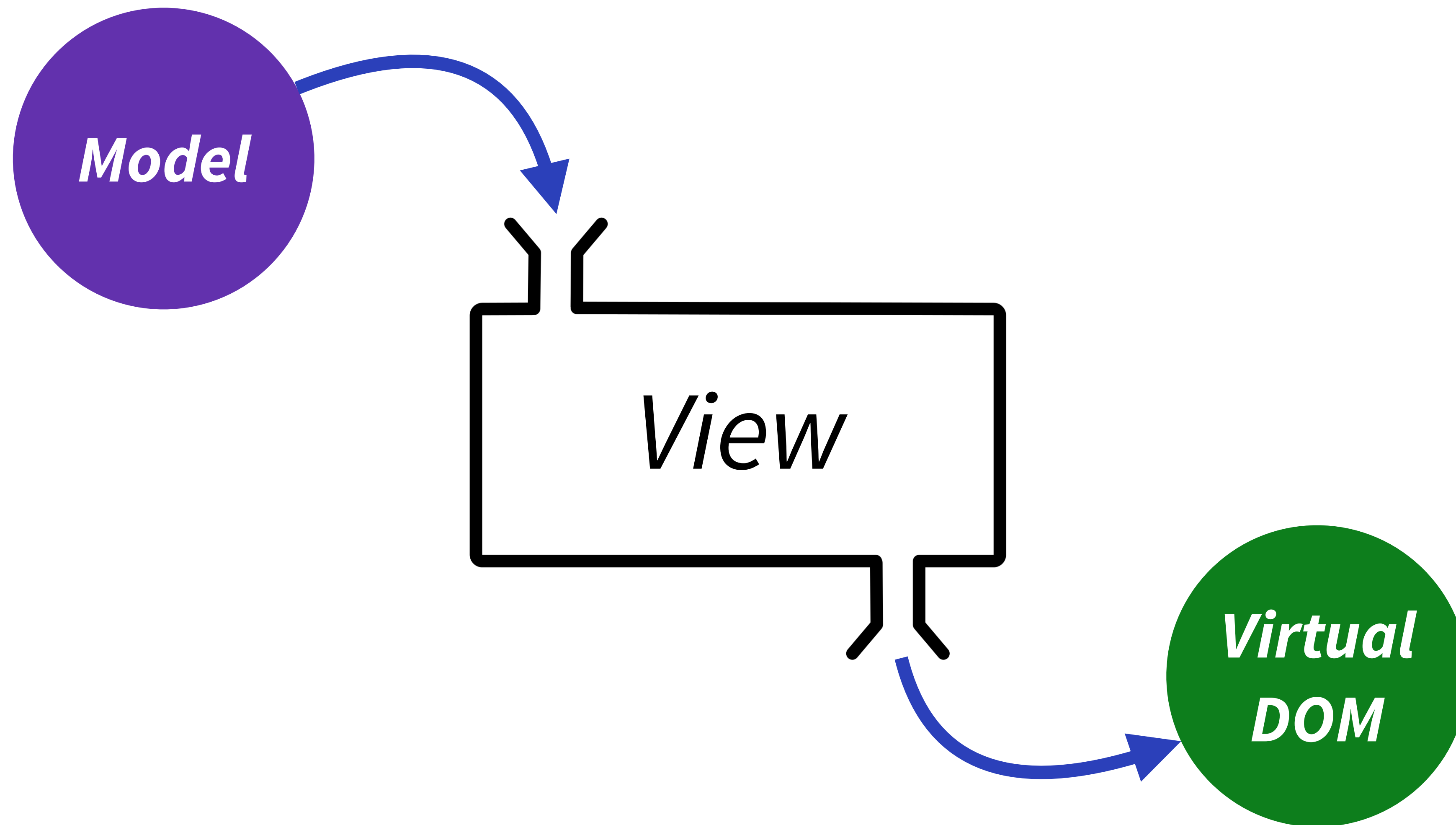
The Elm Architecture

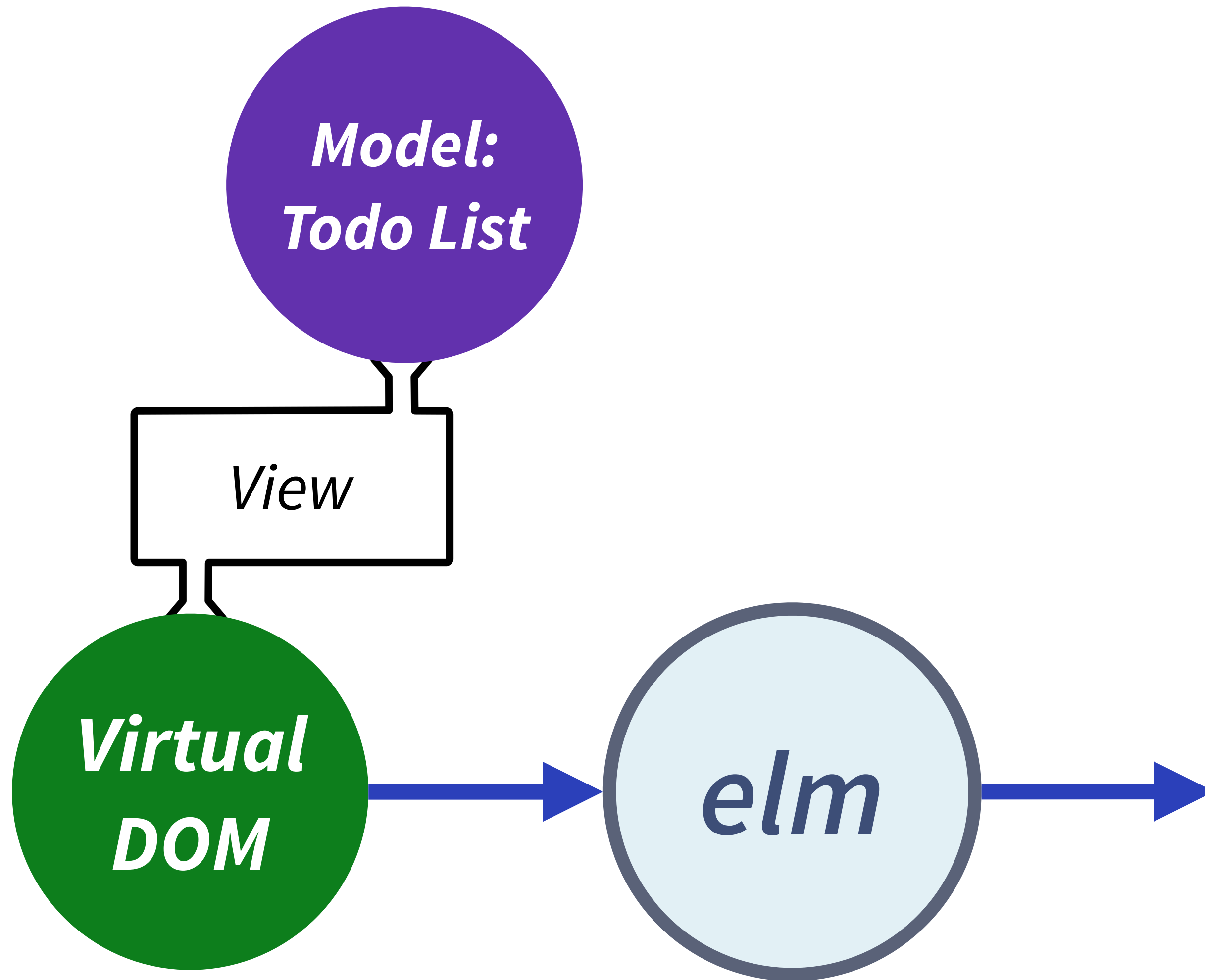




Application
State

UI as a Function





<html />

Todo List

Learn Elm

Build awesome Elm apps

Learn functional programming

*User deletes
first todo item*



*Model:
New Todo
List*

<html />

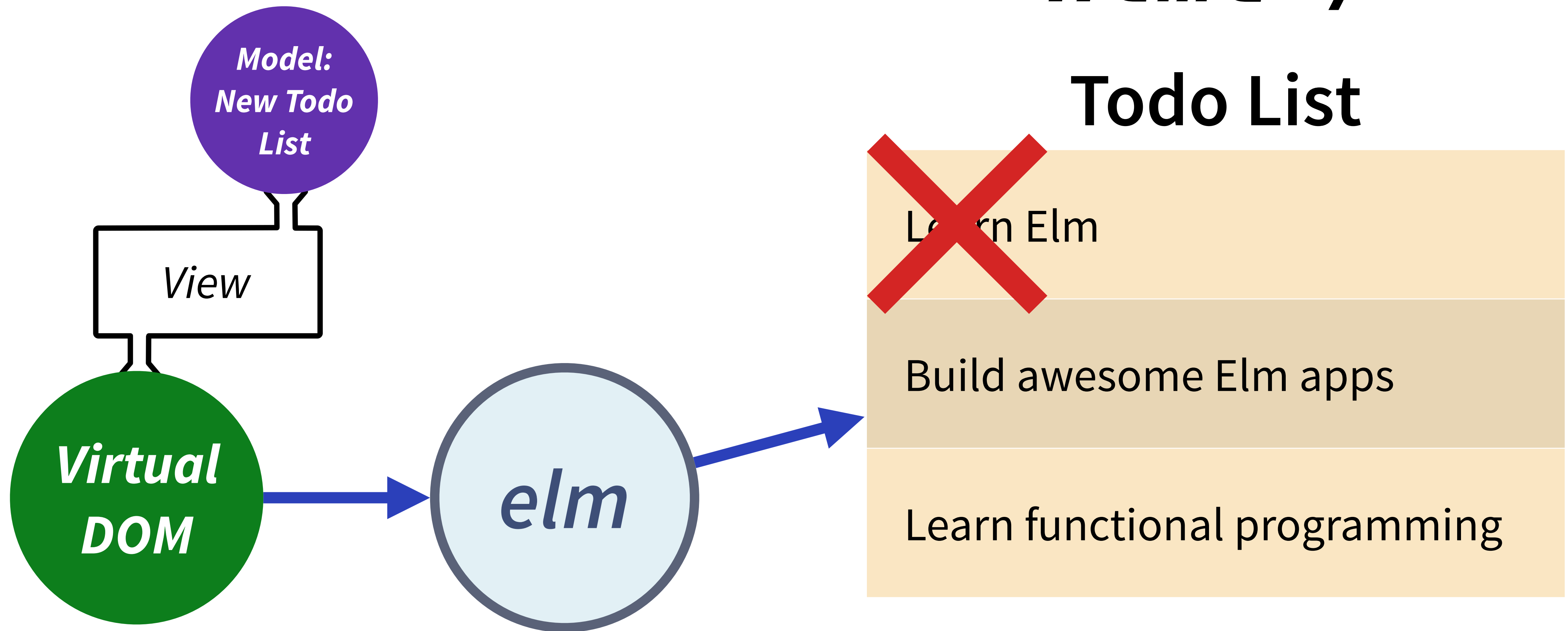
Todo List

Learn Elm

Build awesome Elm apps

Learn functional programming

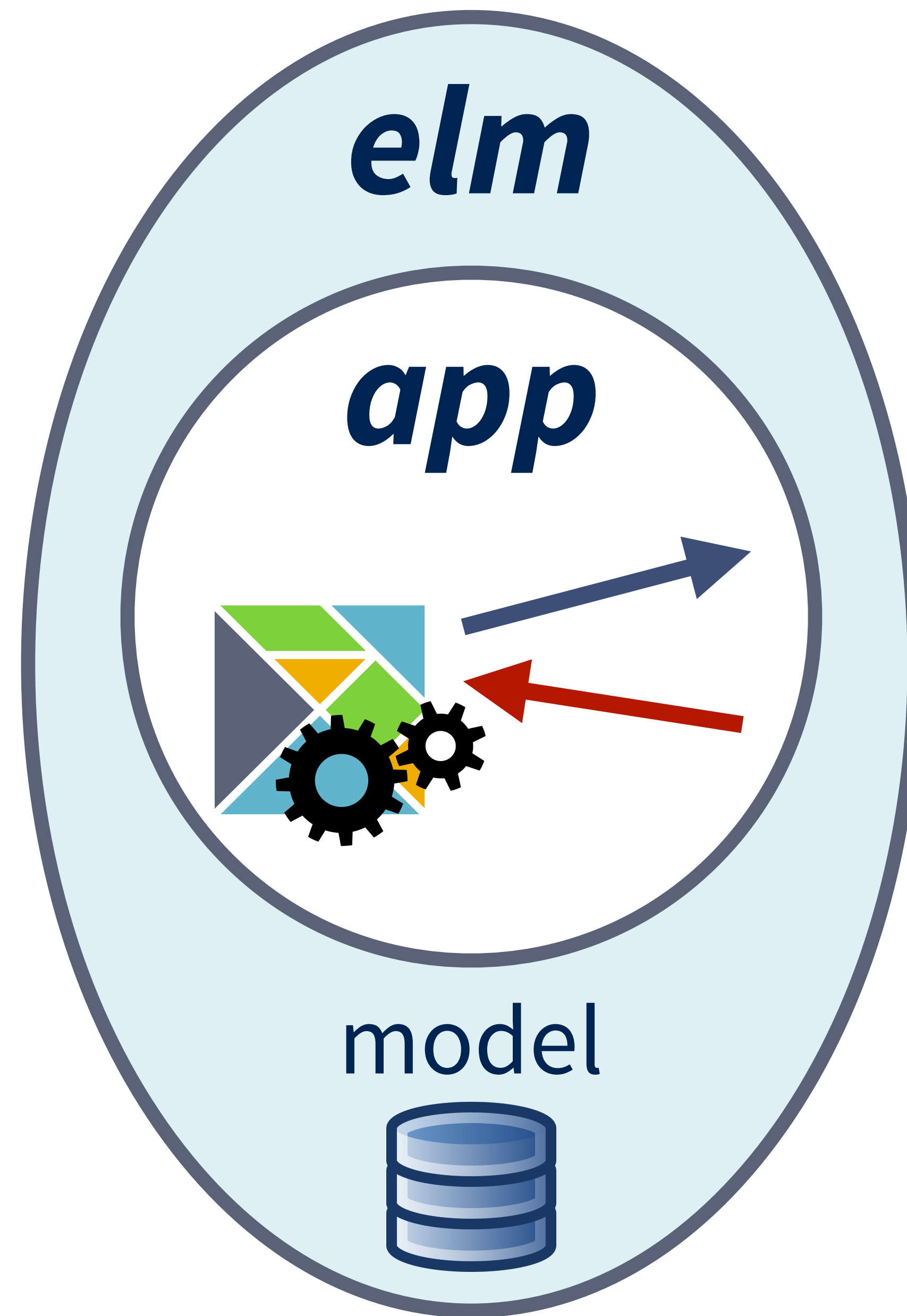
Elm notices missing todo item in virtual DOM list, so it just removes the corresponding `` instead of fully rendering.



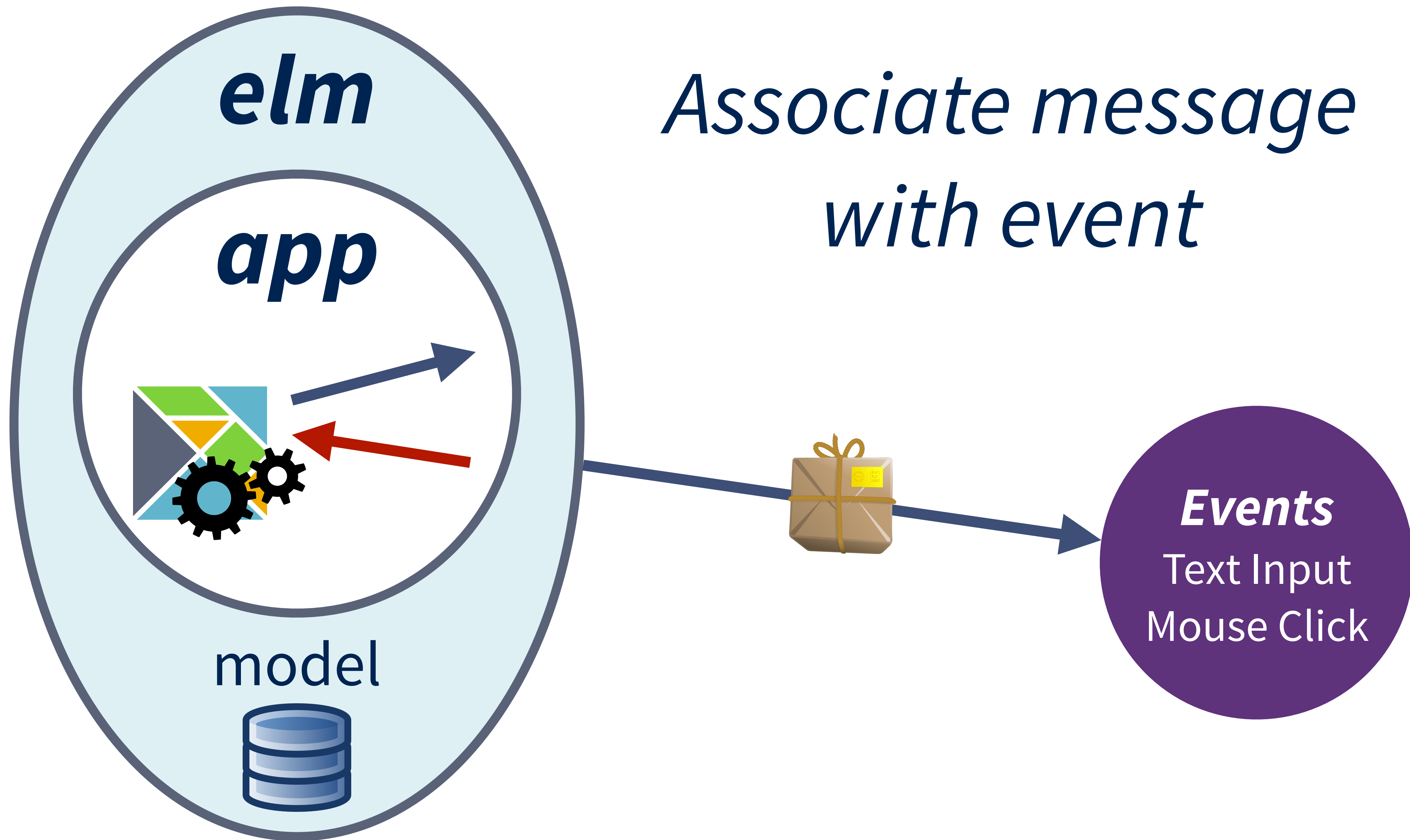


Messages

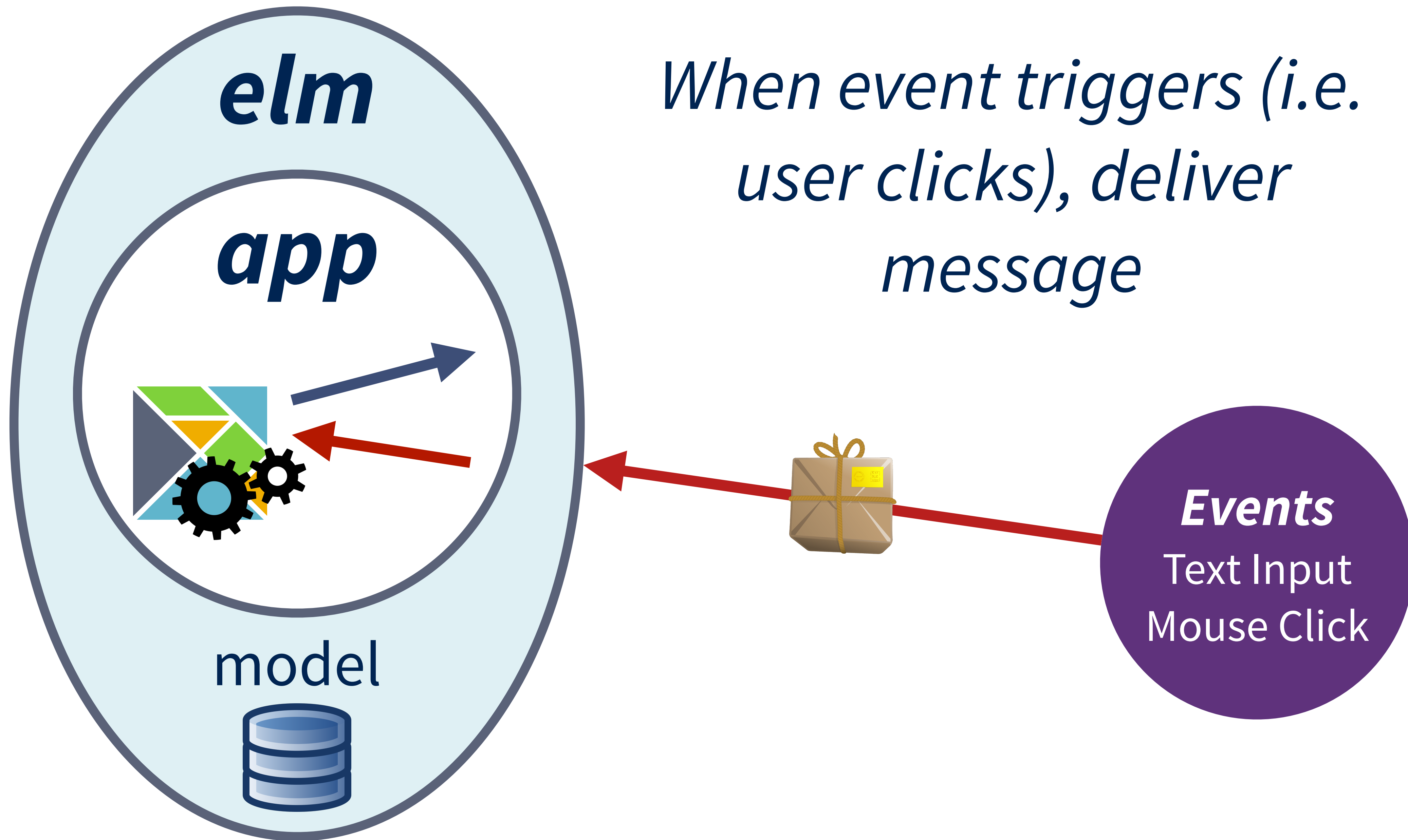
Standardized
application events

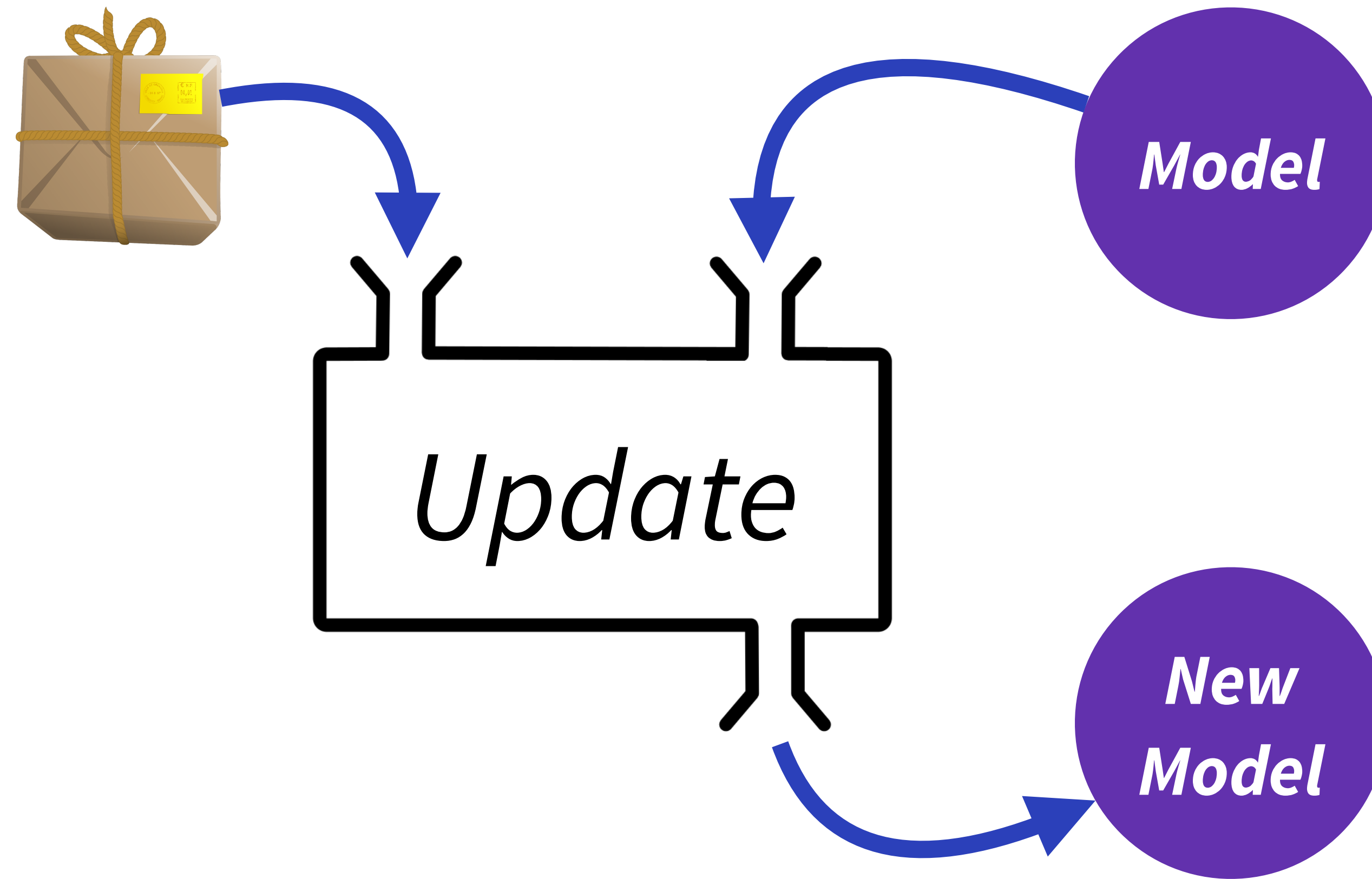


*Associate message
with event*



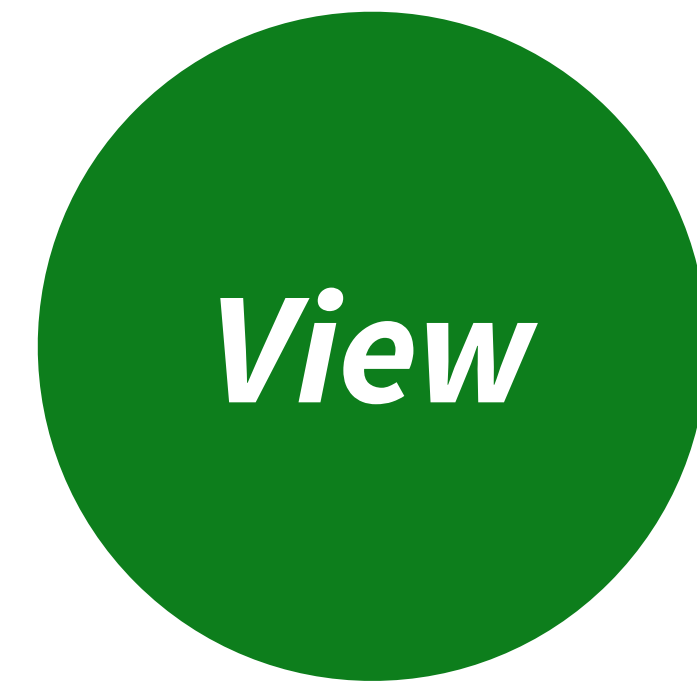
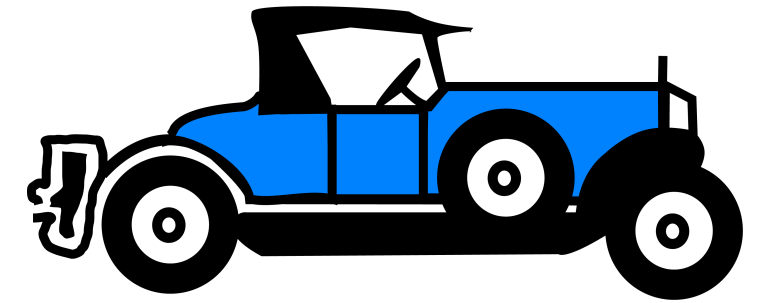
*When event triggers (i.e.
user clicks), deliver
message*



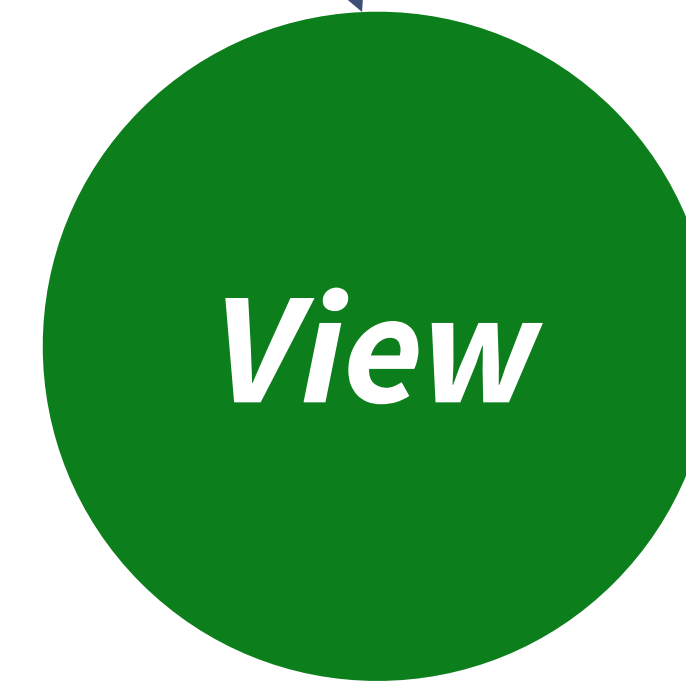
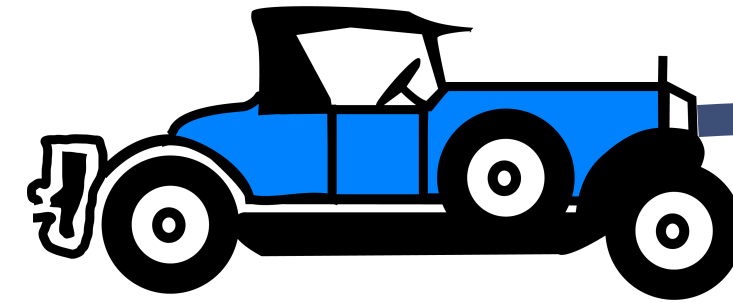


*Respond to messages and
create new state*

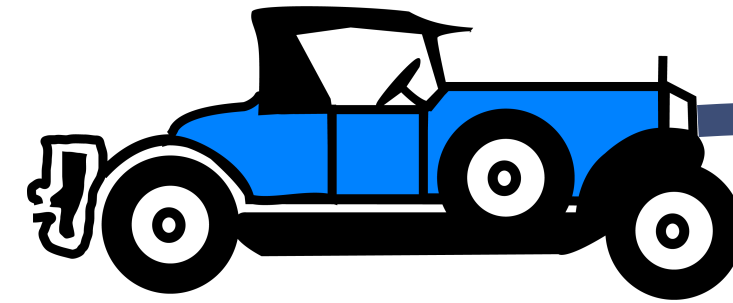
model



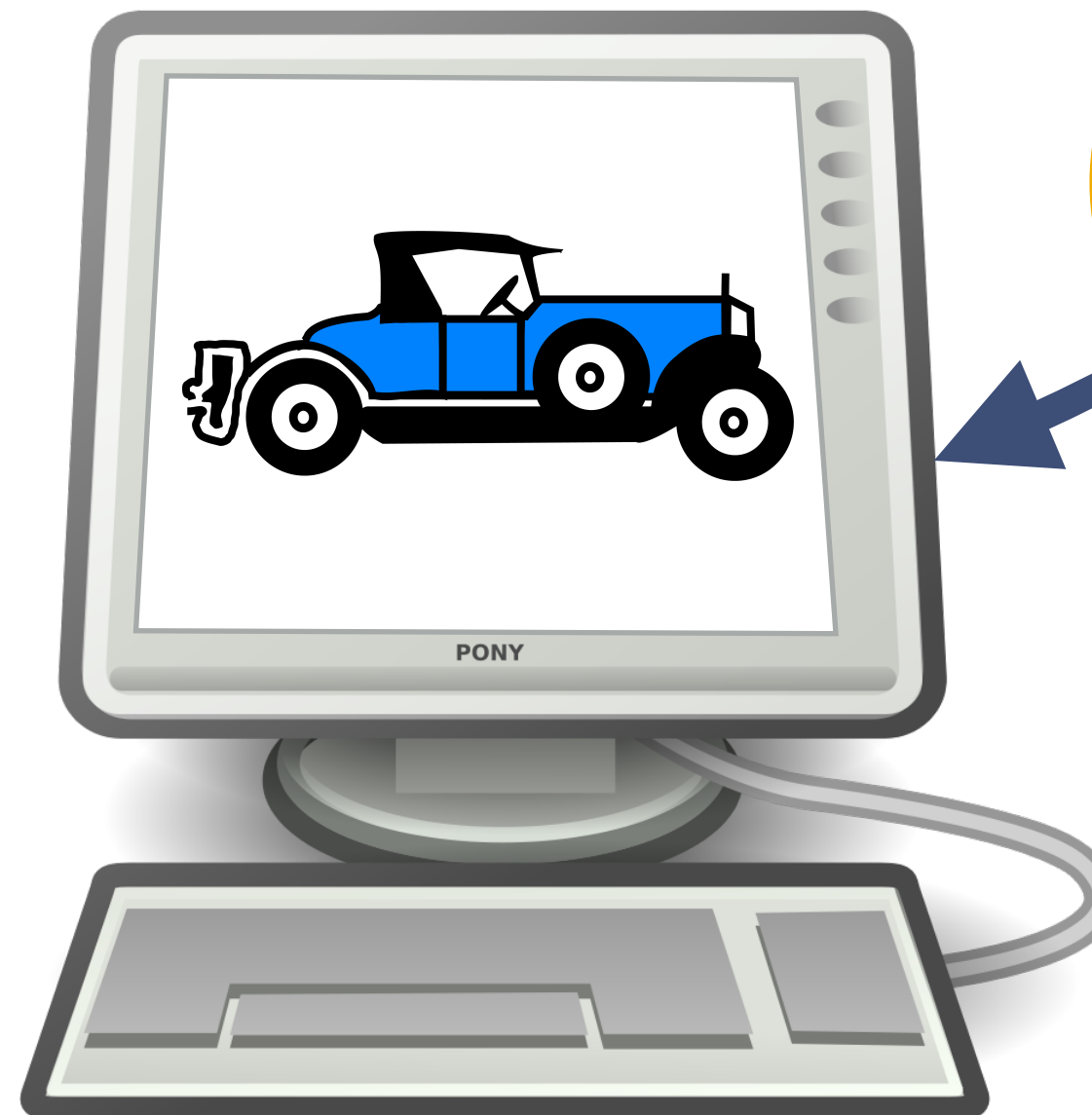
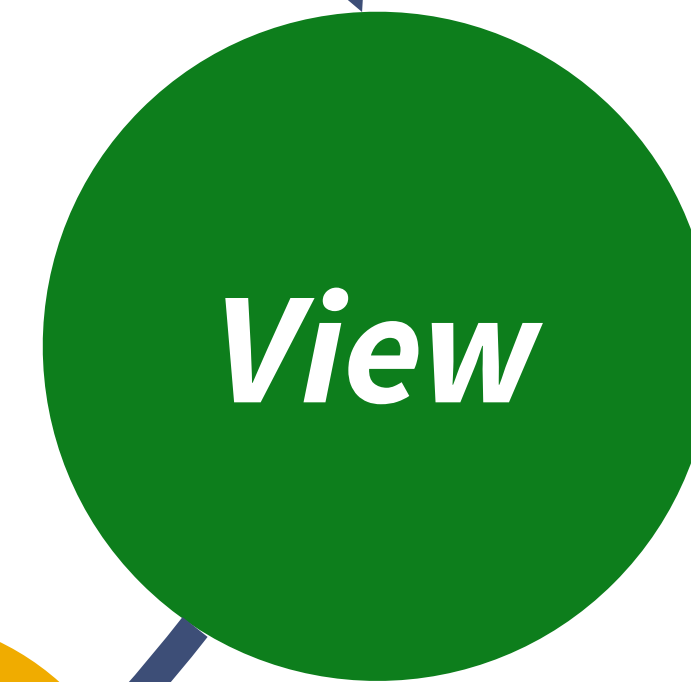
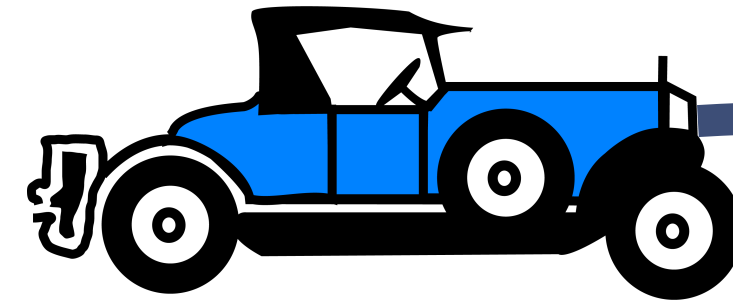
model



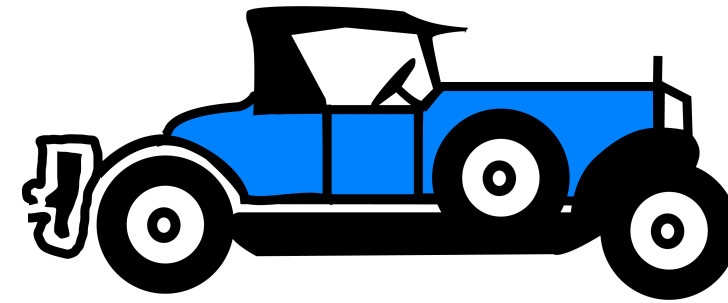
model



model

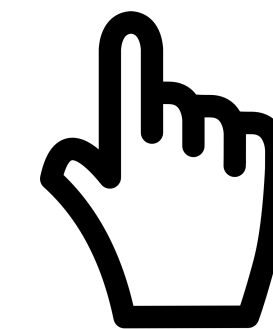


model



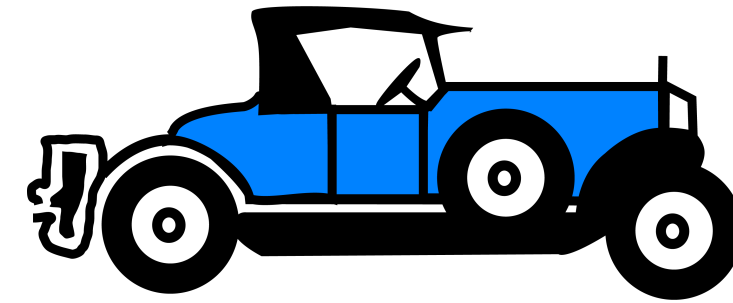
Update

View



Select red color

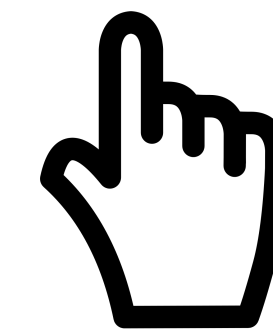
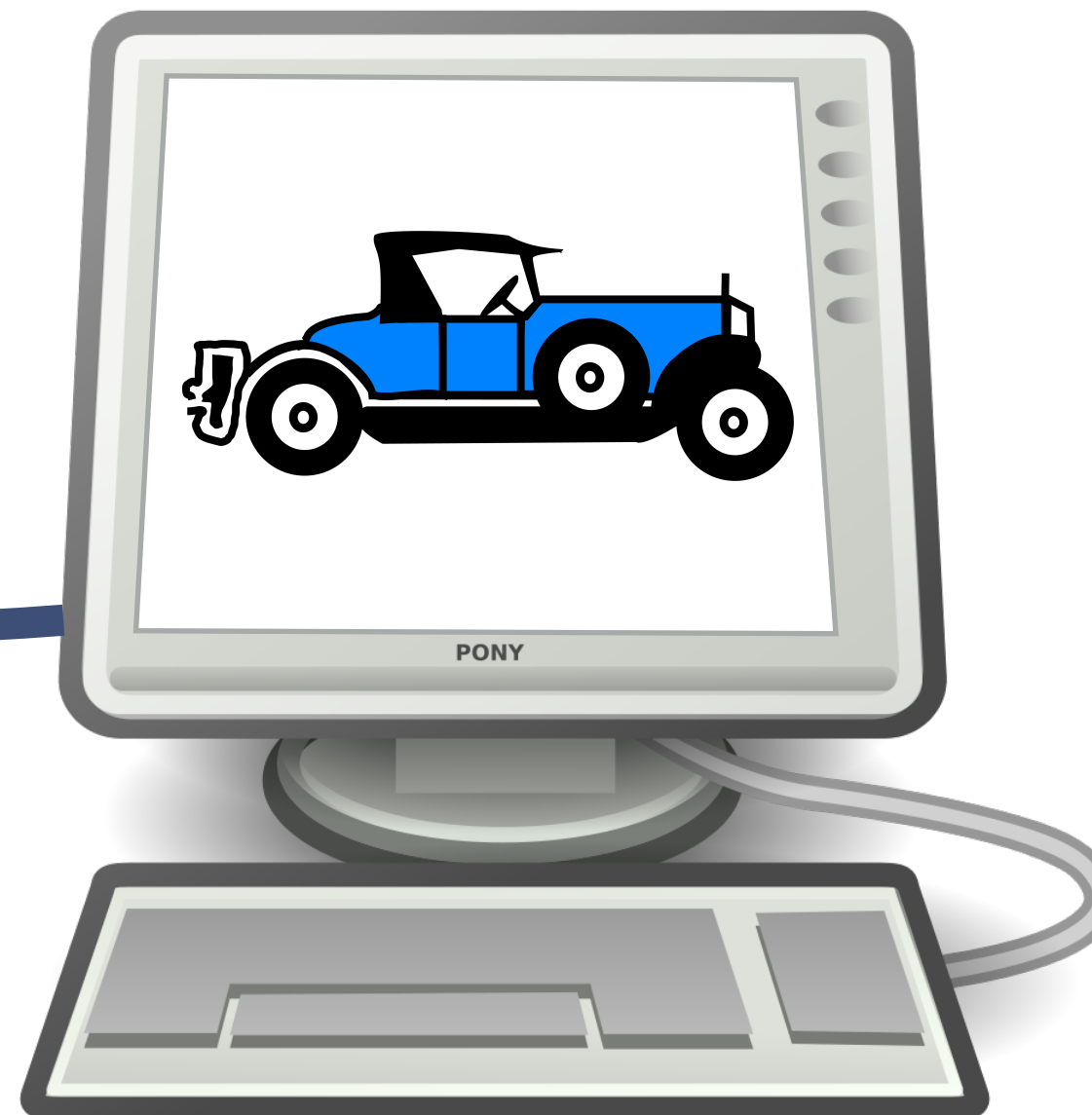
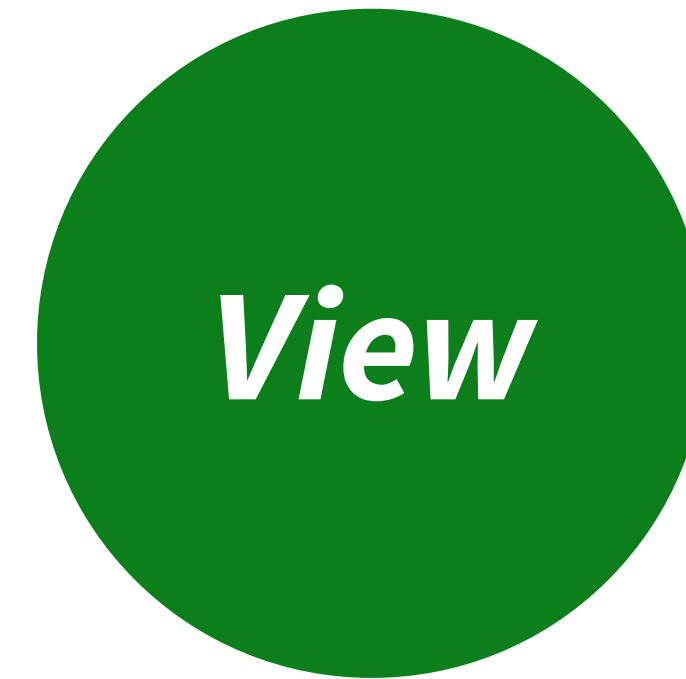
model



Update

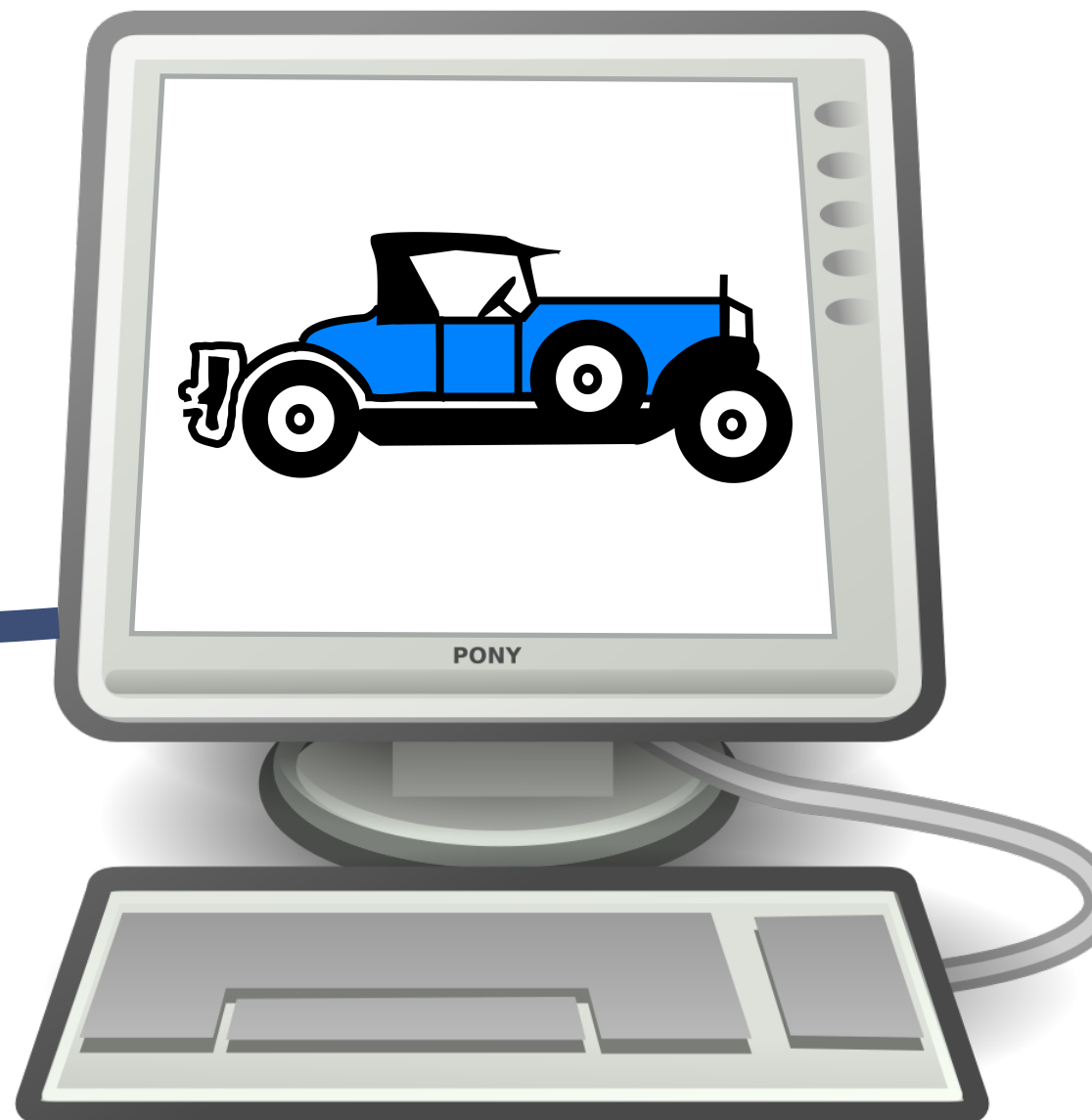
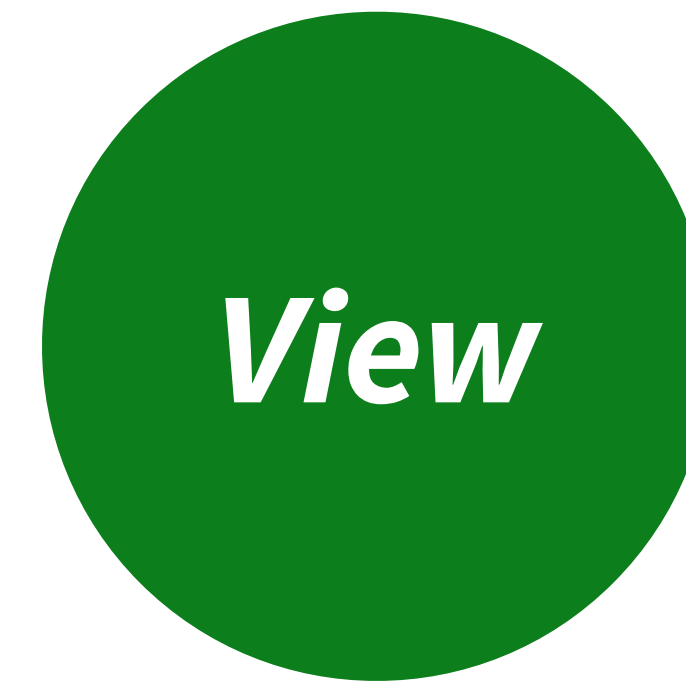
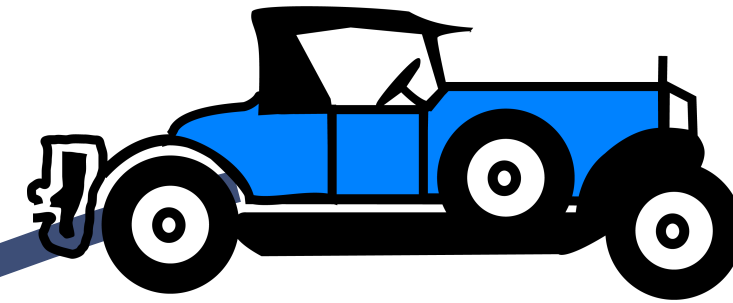


View



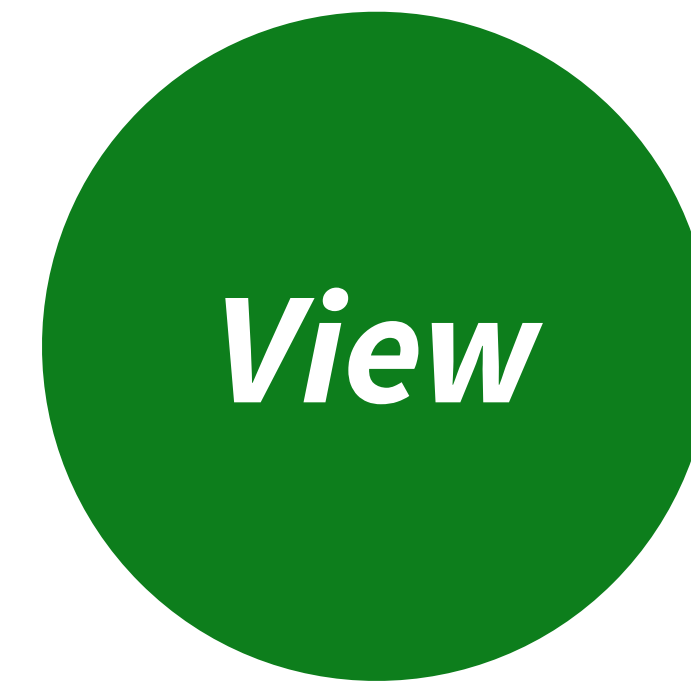
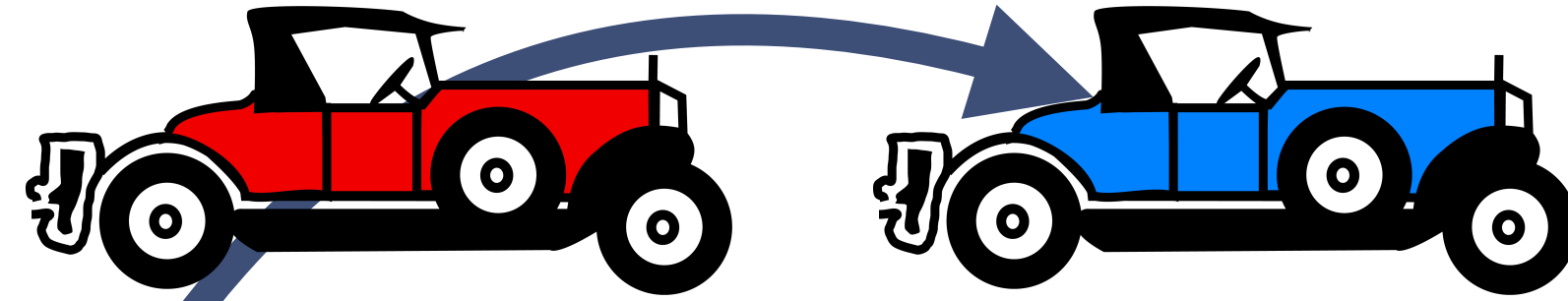
Select red color

model

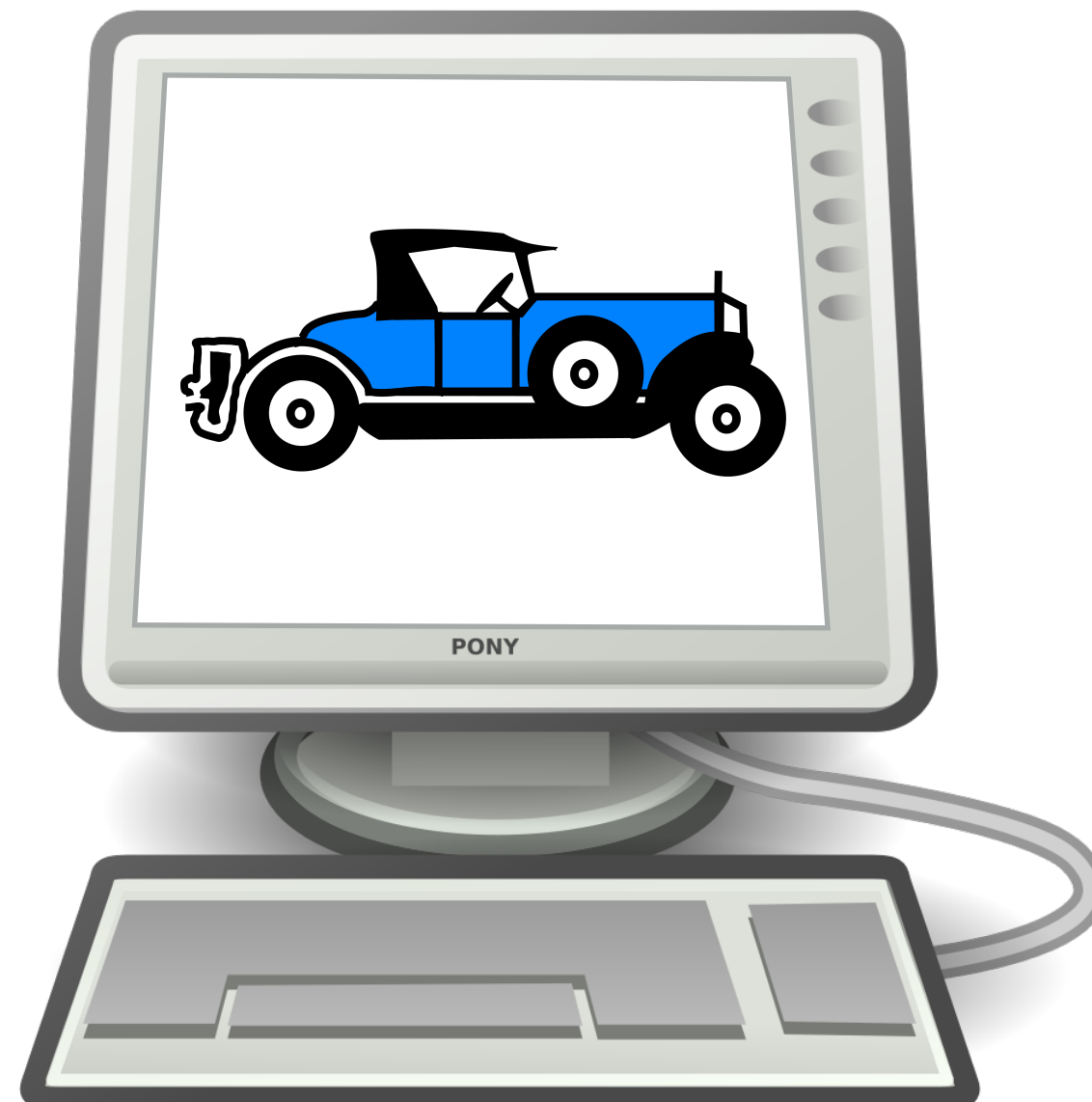
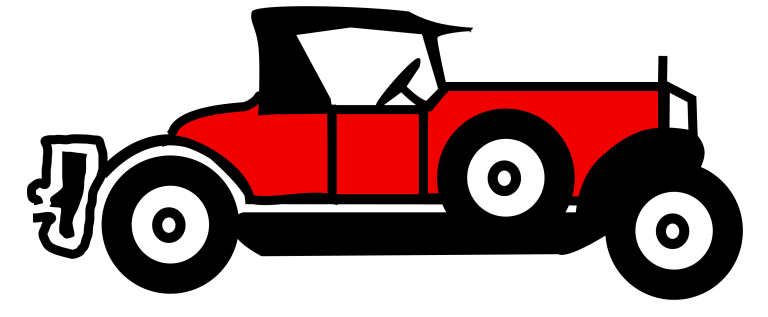


Select red color

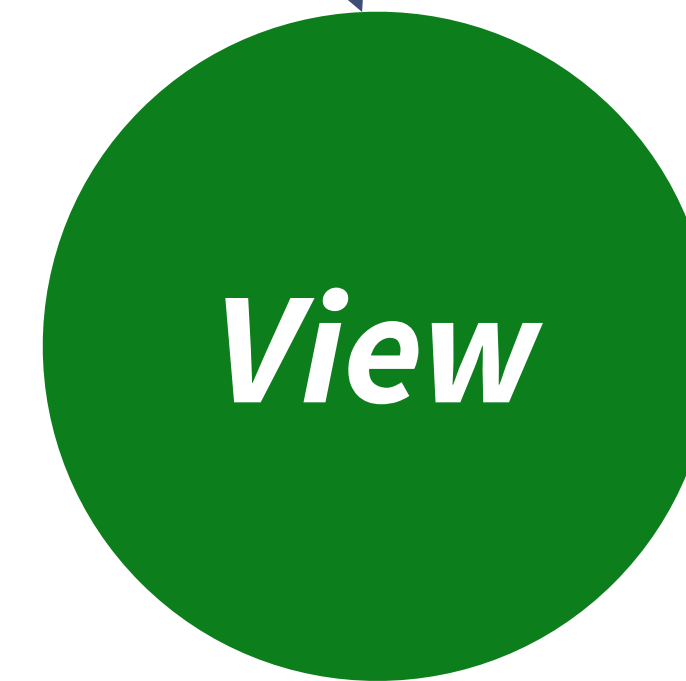
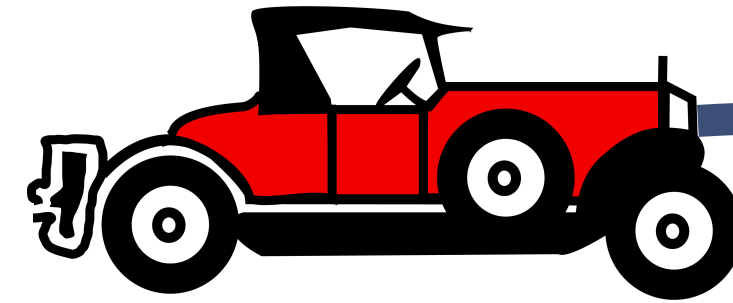
model



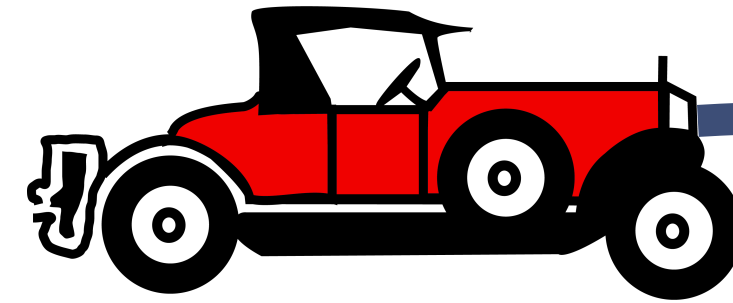
model



model



model



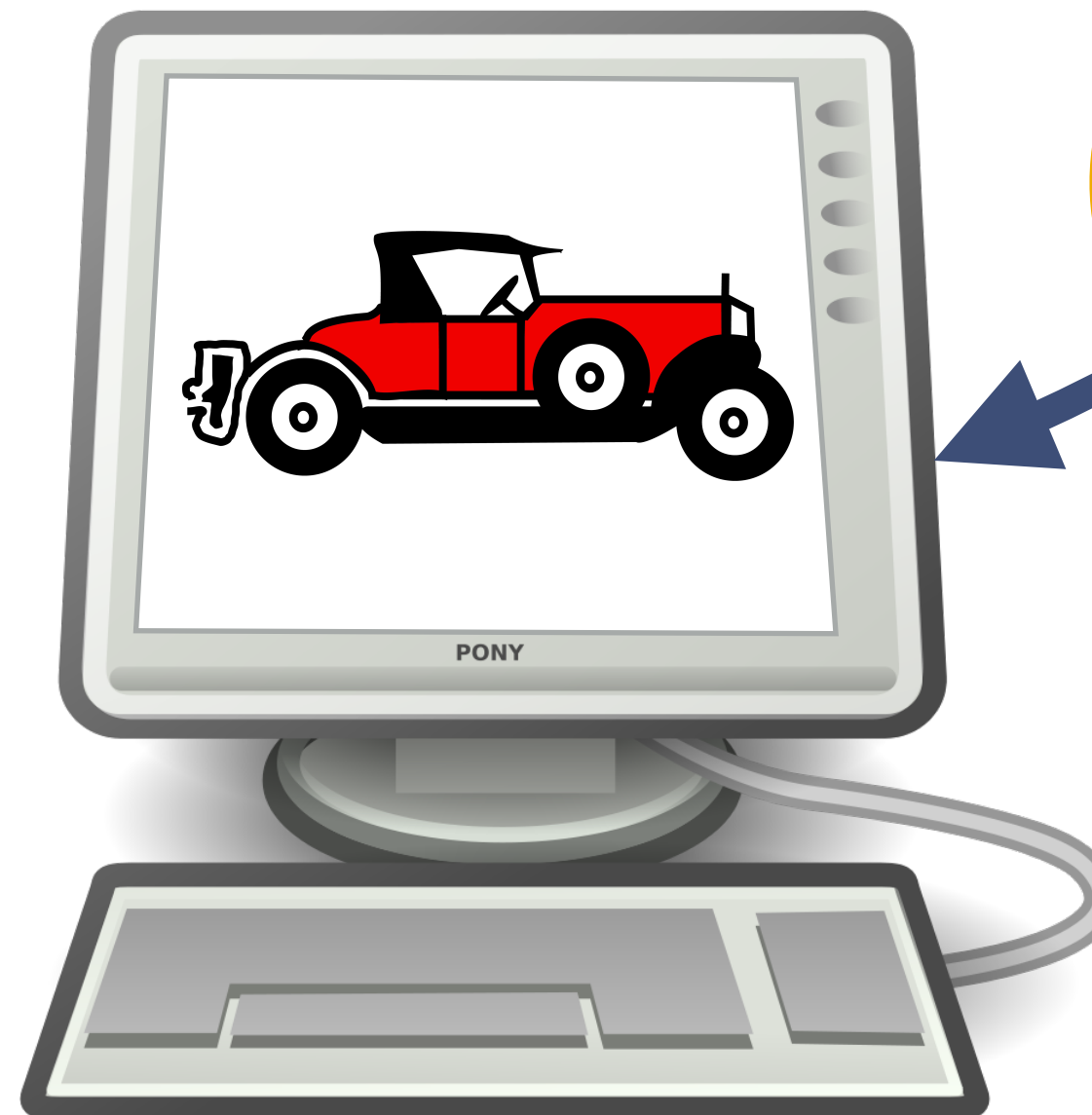
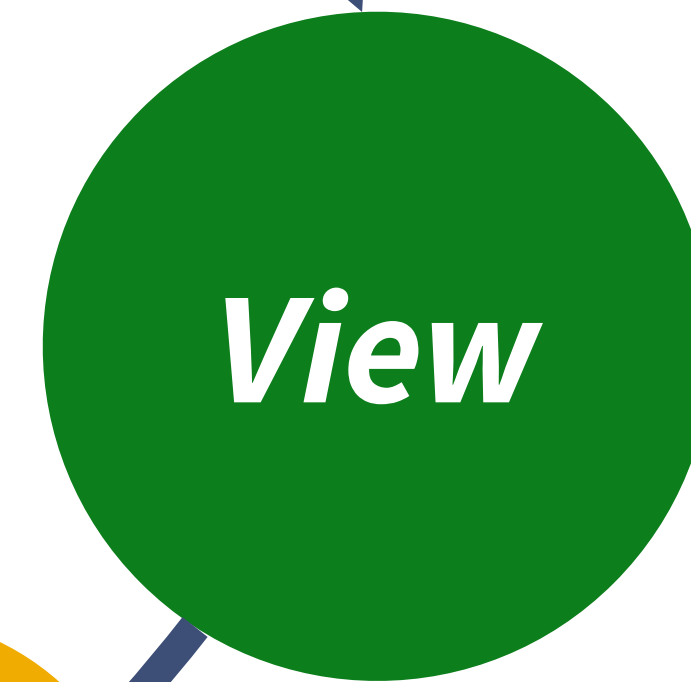
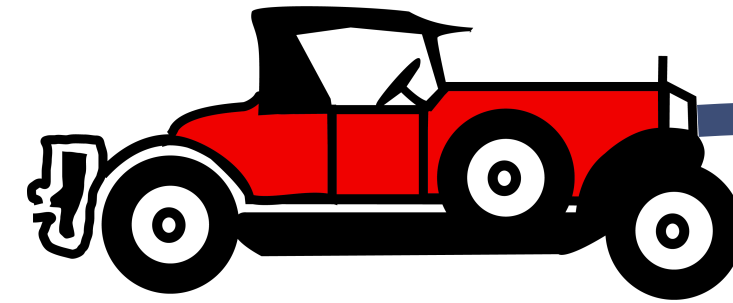
Update

View

VDOM



model



Coding Time!

Resources

- bit.ly/programming-elm
- elm-lang.org
- elm-lang.org/examples
- guide.elm-lang.org
- www.elm-tutorial.org
- builtwithelm.co
- *Slack*
 - elmlang.herokuapp.com

Thanks!

Slides: bit.ly/fluent-elm-slides-18

Repo: bit.ly/fluent-elm-18

Jeremy Fairbank

 *@elpapapollo* /  *jfairbank*