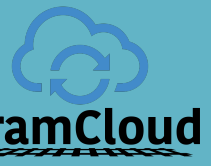
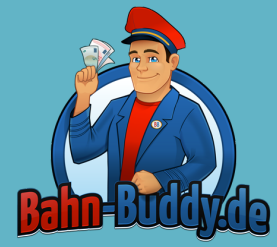
THE FUTURE OF FRONT-END DEVELOPMENT

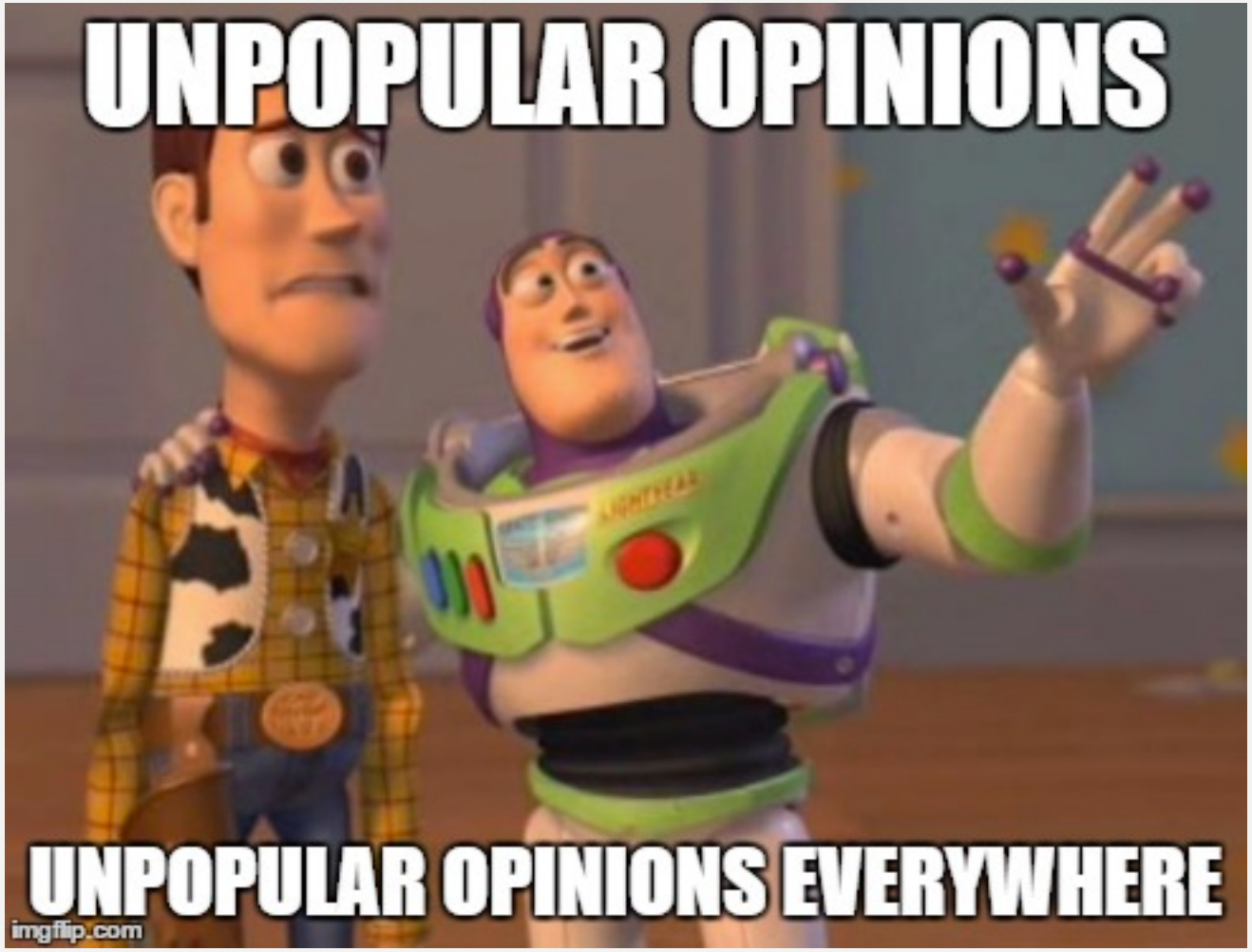
ALEXANDER THIEMANN

BOB 2017

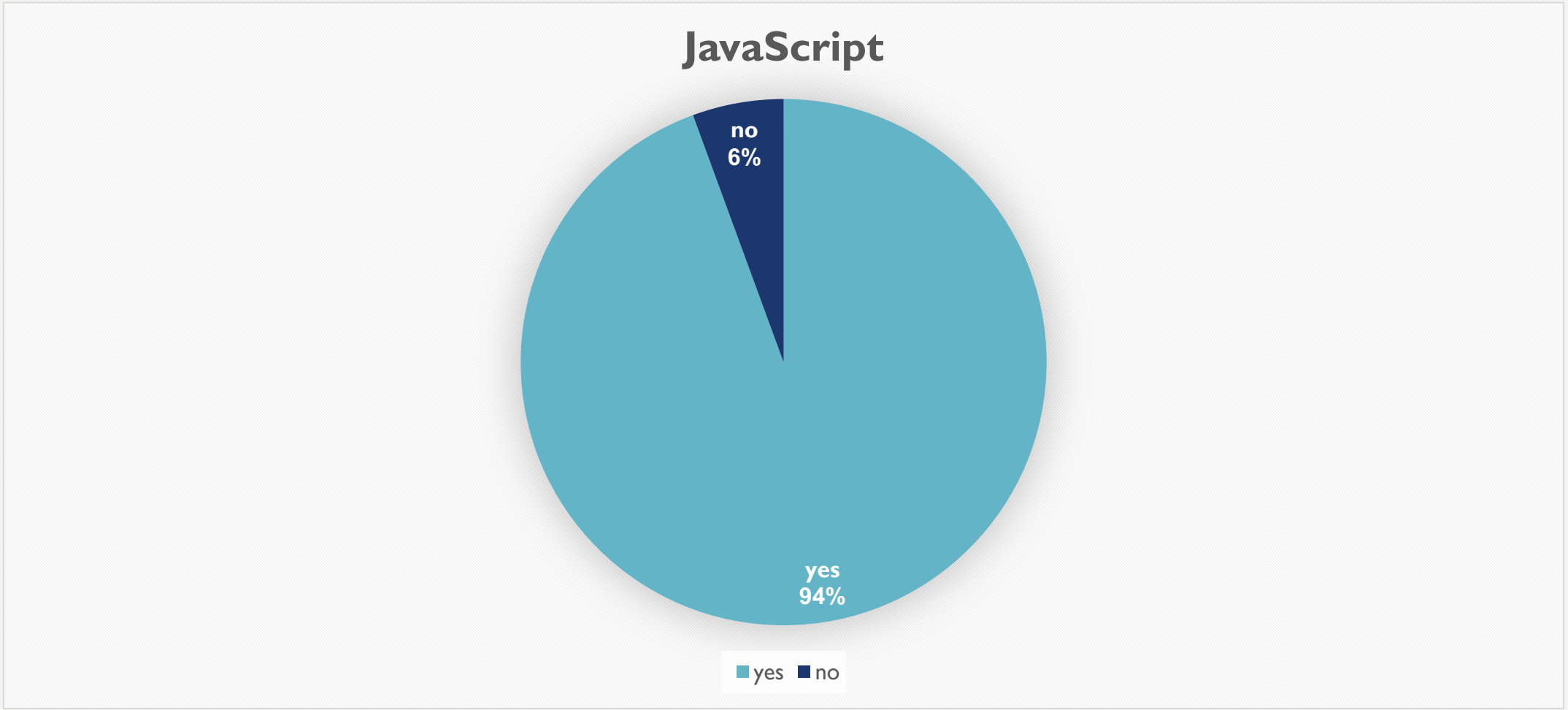




THIS TALK


UNPOPULAR OPINIONS
UNPOPULAR OPINIONS EVERYWHERE
imgflip.com

WEB FRONT-END



JAVASCRIPT

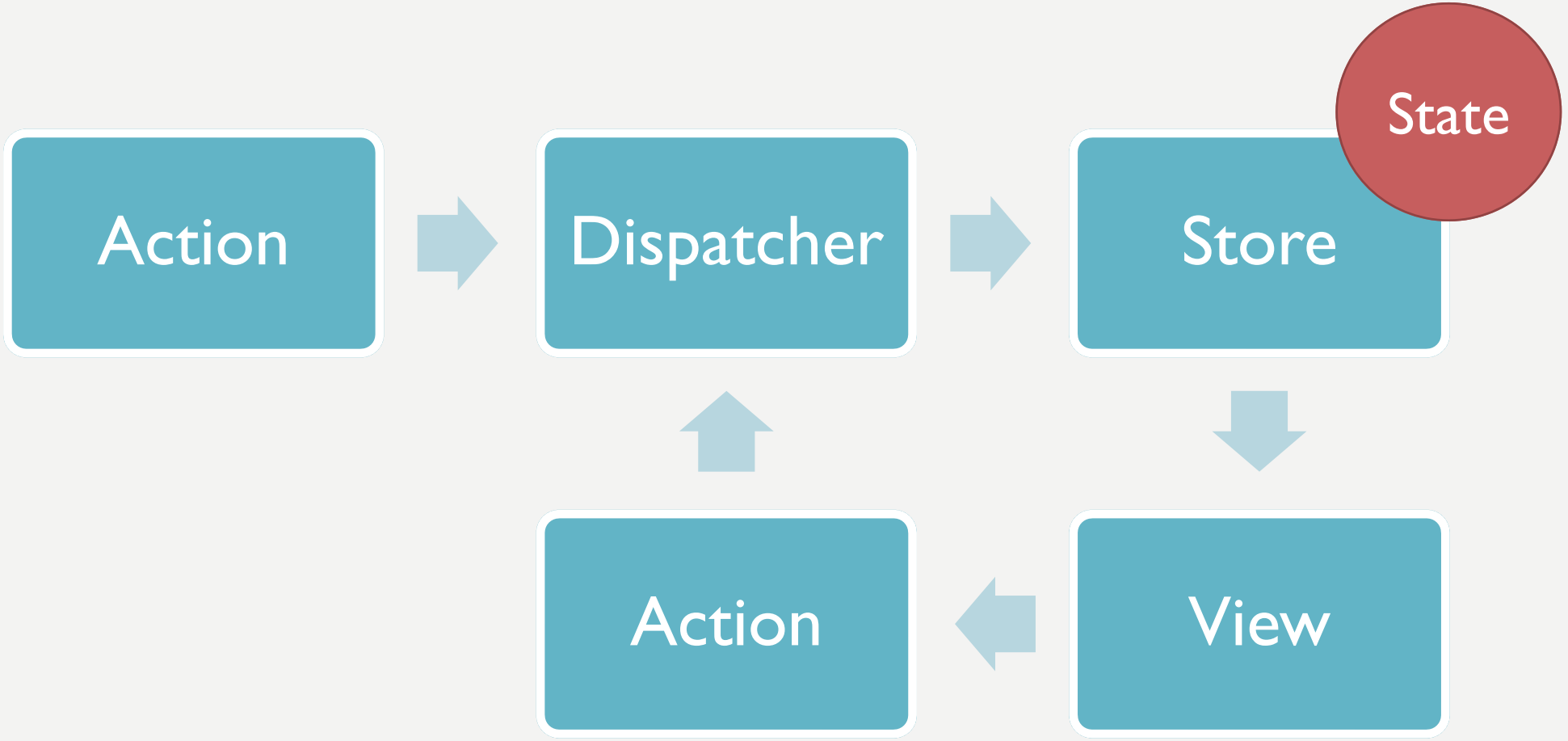
```
NaN == NaN; // true or false?
NaN === NaN; // true or false?
isNaN(NaN); // true of false??
```


JavaScript
The Definitive Guide
David Flanagan
O'REILLY
JavaScript:
The Good Parts
Douglas Crockford
O'REILLY
YAHOO! PRESS

FRONT-END “FRAMEWORKS”

- React
- VueJS
- AngluarJS
- Ember.js

FLUX PATTERN



FLUX PATTERN: STORE

```
applyAction :: Action -> State -> State

renderView ::
    (Action -> Effect ())
    -> State
    -> Html
```

JAVASCRIPT LIBRARIES

- Object.freeze()
- ImmutableJS: “Immutable collections for JavaScript”
- Ramada: “Practical functional JavaScript”
- RxJS: “A reactive programming library for JavaScript”

HOW CAN WE (TRY TO) ENFORCE?

```
test/fixture/1.js
  line 1  col 0  error  'alert' is not defined.
  line 1  col 0  error  Unexpected alert.
  line 1  col 0  error  Strings must use doublequote.

test/fixture/2.js
  line 1  col 0  error  Identifier 'hello_worl' is not in camel case.
  line 1  col 0  error  Strings must use doublequote.
  line 1  col 0  error  hello_worl is defined but never used

✖ 6 problems
```

TYPE CHECKERS

- Google Closure Compiler
- Facebook Flow
- TypeScript

TYPESCRIPT

```
class Greeter {
    constructor(public greeting: string) { }
    greet(): string {
        return "<h1>" + this.greeting + "</h1>";
    }
}

const greeter = new Greeter("Hello, world!");
document.body.innerHTML = greeter.greet();
```

TYPESCRIPT

- “JavaScript with type system and syntax sugar”
- Steady development by Microsoft (Frequent releases!)
- Small learning curve
- Many (prominent) adopters
  - AngularJS
  - SKY
  - Ubisoft
  - Checkpad MED (hi!)
- Great ecosystem (all of npm with many .d.ts files)

iPad

KAR-P

Patienten 0-9 | Patienten A-Z

Aufgaben Station

Befunde

Bewegungen

ZI: 1

Oppermann, Vera ♀
15.08.1962 (54) Zi: 1a
18.09.2016, 05:00 (4)
teamA
#99881999 / Allianz (P)

leeres Bett b

ZI: 2

leeres Bett a

leeres Bett b

Oppermann, Vera · #99881999 · KAR-P, Zi: 1a

WORKFL

AKTE

11:50

100 %

Verlaufsdokumenta... Bearbeiten

VERLAUF

22.09.2016, 11:49 Uhr
Schmerzen sind jetzt besser

22.09.2016, 11:47 Uhr
Patientin klagt über starke Schmerzen

LETZTE ÄNDERUNG

Editiert von Lohmann am 22.09.2016, 11:49 Uhr

22.09.2016, 11:49 Uhr

VERLAUFSEINTRAG

Schmerzen sind jetzt besser

WER UND WANN?

Benutzer Lohmann

Datum / Zeit 22.09.2016, 11:49 Uhr
heute

CHECKPAD'S TYPESCRIPT STORY

- Converted from plain JS (with some Closure annotation) in 1-2 weeks
- Now 55 000 lines of code
- In general, very happy with results
  - New checking features often find more edge cases (noImplicitThis)
  - Turn checking features on or off
    - Not gradually 🤔
    - Effort to fix existing code

NO IMPLICIT THIS

```
ClickHandler.prototype.clearShoppingCart = //...
ClickHandler.prototype.fillShoppingCart = // ...
ClickHandler.prototype.handleClick = function () {
  this.clearShoppingCart();
  this.timer = window.setTimeout(function() {
      this.fillShoppingCart(['Apple']);
      // ^ Runtime: undefined is not a function
  }, 20);
};
```

NO IMPLICIT THIS

```
ClickHandler.prototype.handleClick = function () {
  this.clearShoppingCart();
  // ^ this has any type
  this.timer = window.setTimeout(function() {
  // ^ this has any type
      this.fillShoppingCart(['Apple']);
      // ^ this has any type
  }, 20);
};
```

NO IMPLICIT THIS

```
class ClickHandler {
    private timer: number = 0;
    clearShoppingCart() {}
    fillShoppingCart(x: string[]) {}
    handleClick(this: ClickHandler) {
        this.clearShoppingCart();
        this.timer = window.setTimeout(() => {
            this.fillShoppingCart(['Apple']);
        }, 20);
    }
}
```

```
const ch = new ClickHandler();
document.onclick(ch.handleClick);
// ^ type error!
```

NO IMPLICIT THIS

```
const ch = new ClickHandler();
document.onclick(ch.handleClick);
// ^ type error!
```

TYPESCRIPT

PRO	"CON"
• Mature compiler • Stable language • Steady development • Great ecosystem • Great tooling and editor support • Reasonably fast compiler • Good type system, continuously improving • Prevents many common JavaScript errors	• Checking flags are "all or nothing" • No immutability • No side effect control

ELM

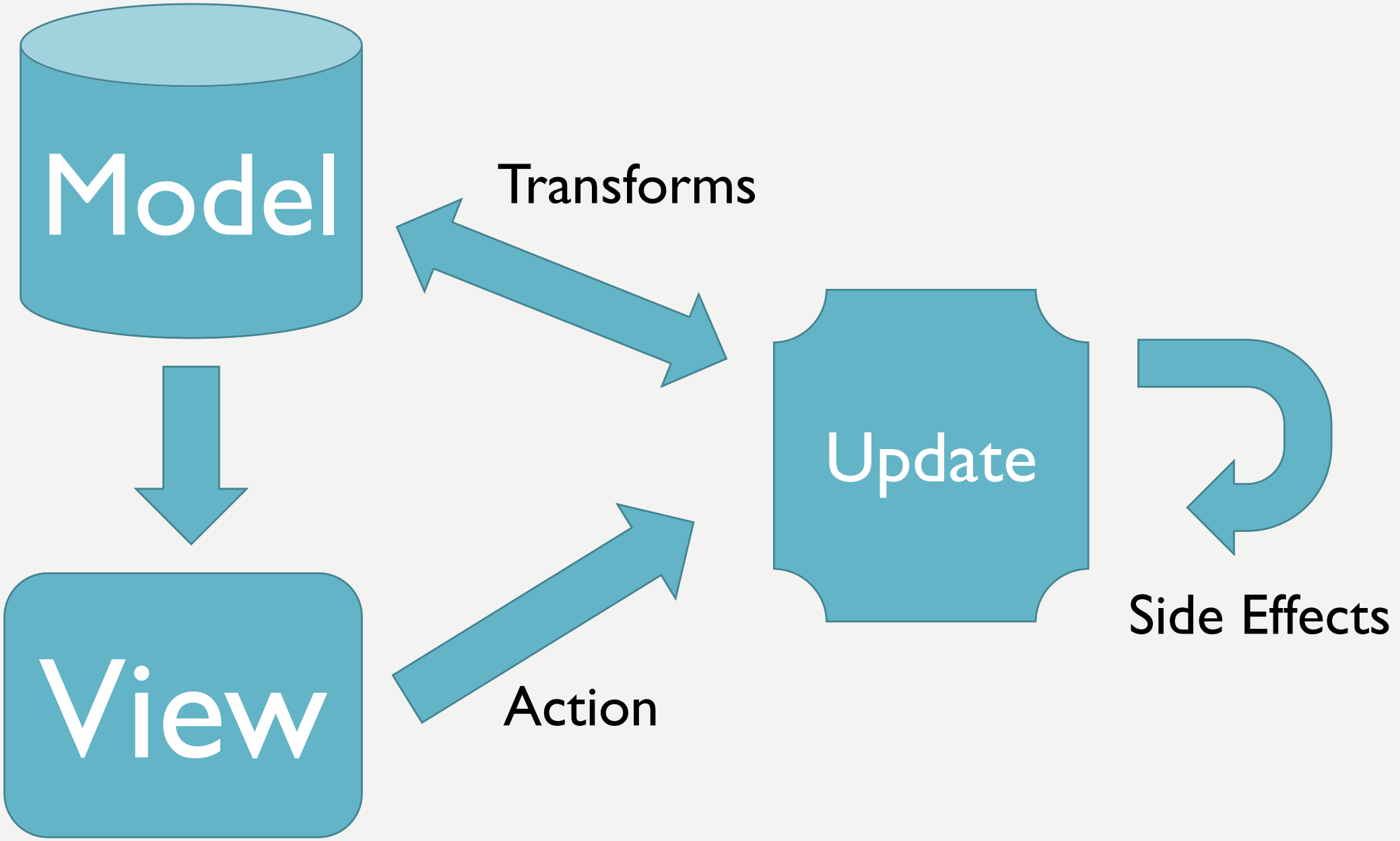
```
import Html exposing (beginnerProgram, div, button, text)
import Html.Events exposing (onClick)

main =
  beginnerProgram { model = 0, view = view, update = update }

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

type Msg = Increment | Decrement

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


Model
Transforms
Update
Side Effects
View
Action

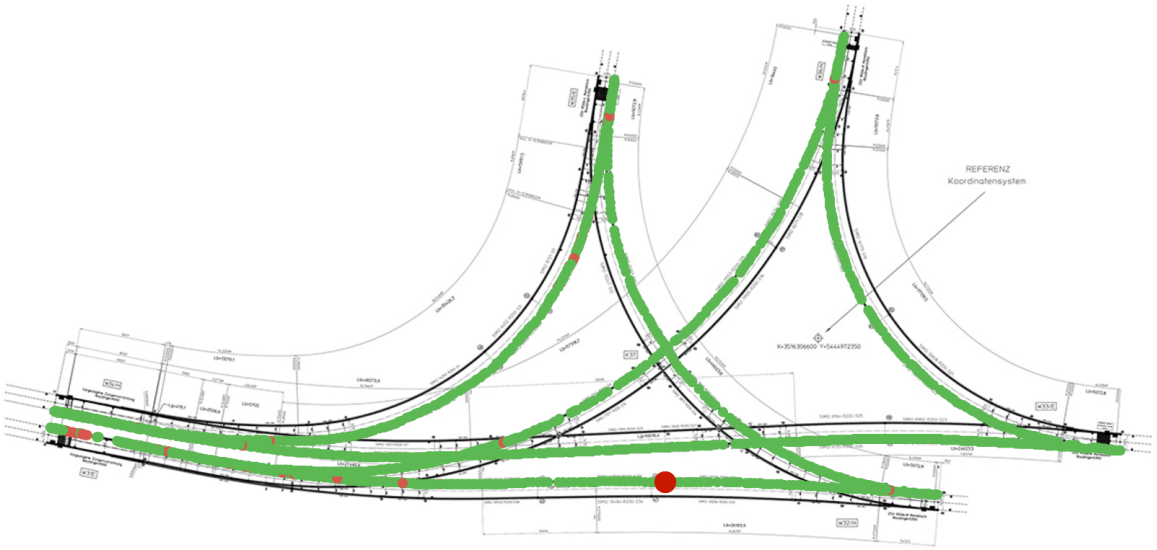
ELM

- ML / Haskell like language, simpler type system and smaller syntax (e.g. no type classes)
- Purity, strong static type system, ...
- Goal: joyful and bug free (web) front-end apps, **easy to learn**
- slower development, manly the creator of language (Evan Czaplicki)
- Average ecosystem
  - Elm packages: ~400
  - wrapping existing JavaScript code is strongly discouraged (undocumented, but works!)
- Used at noredink, Prezi, CircuitHub, BahnBuddy (hi!), TramCloud (hi!) ...

TramCloud

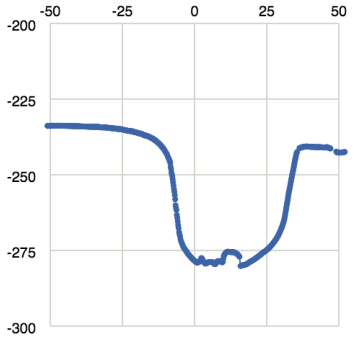
Spurweite	1435 mm
Inbetriebnahme	07.07.2012
Messung	06.08.2015 - 10:12

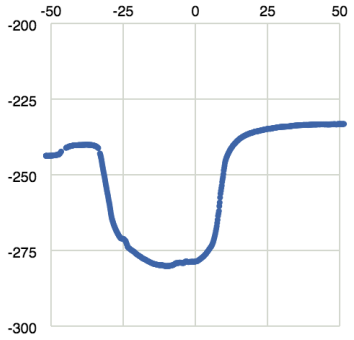
Messung #6



Spurweite: 1436 mm

Rillenbreite	45 mm
Rillentiefe	45 mm

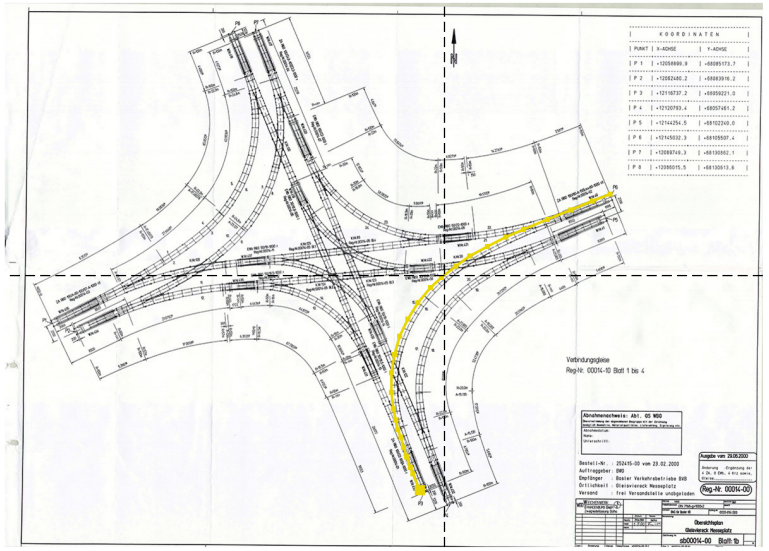


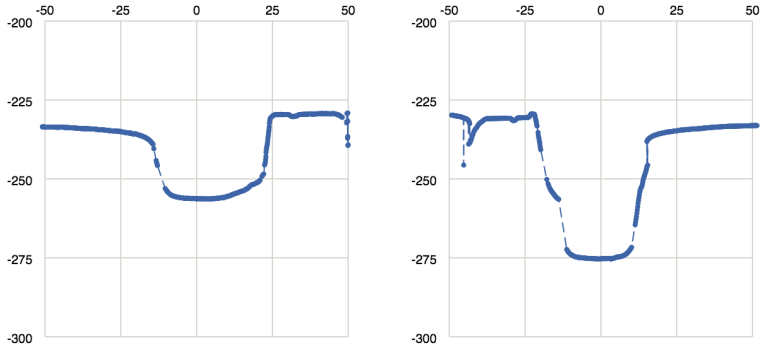


Rillenbreite	45 mm
Rillentiefe	45 mm

TramCloud

Bewegen Sie die Maus über eine Messung in der Tabelle





KR 122 - sha	Spurweite	beide	W0	1001 mm	1005 mm
KR 122 - Wra / Wha	Rillenweite	beide	W0	30 mm	L: 36 mm R: 31 mm
KR 122 - Tha / Tra	Rillentiefe	beide	W0	16 mm	L: 15 mm R: 15 mm
KR 122 - Whda / Wrda	Rillenweite	beide	W0	30 mm	L: 31 mm R: 41 mm
KR 122 - sda	Spurweite	beide	W0	1001 mm	1005 mm
KR 122 - sdc	Spurweite	beide	W0	1001 mm	1005 mm
KR 122 - Whdc / Wrdc	Rillenweite	beide	W0	30 mm	L: 36 mm R: 32 mm
KR 122 - shc	Spurweite	beide	W0	1003 mm	1005 mm
KR 122 - Whc	Rillenweite	links	W0	33 mm	L: 42 mm
KR 122 - Wrc	Rillenweite	rechts	W0	33 mm	R: 41 mm
KR 122 - sec	Spurweite	beide	W0	1002 mm	1009 mm
KR 030 - sed	Spurweite	beide	W0	1000 mm	1004 mm
KR 030 - shd	Spurweite	beide	W0	999 mm	1004 mm
KR 030 - Whd / Wrd	Rillenweite	beide	W0	28 mm	L: 36 mm R: 31 mm
KR 030 - Whdd / Wrdd	Rillenweite	beide	W0	28 mm	L: 36 mm R: 34 mm
KR 030 - sdd	Spurweite	beide	W0	999 mm	1007 mm
KR 030 - Thb	Rillentiefe	links	W0	16 mm	L: 15 mm

TramCloud

TRAMCLOUD'S ELM STORY

- Original Front-end in JavaScript + React
- Hacked together quickly and hard to maintain
- Modular approach: each front-end view independent
- August 2015: Try out Elm (0.15) in one component - Crossover management
- Initial development: fun and quick
- Coming from Haskell:
  - Limited ways to abstract
  - Boilerplate code
  - No code sharing with Haskell backend or existing JavaScript code

TRAMCLOUD'S ELM STORY II OR: CUSTOMERS!

- First production use
  - Customer had IE9
  - White page
  - Easy to track down problem: polyfill requestAnimationFrame
- Update to Elm 0.16: needed to change 800 lines of 5000 lines, mostly mechanical
- Customer feature request
  - Rendering using core library Graphics.* API (now removed!)
  - Impossible to achieve with the API, not even with "hacky" FFI
  - 😫
- Never updated to Elm 0.17+
  - Core pattern "Elm Architecture" changed!
  - Significant changes in core library and languages since

ELM

PRO	"CON"
• Very easy to learn • Fun language • Good tooling (Atom/Emacs plugins) • Immutability • Side effect control	• Language and ecosystem: Frequent breaking changes • Integration with existing JavaScript tooling/libraries is discouraged • Limited type system • Boilerplate code • No formal process for changes to language (yet?)

GHCJS

```
{-# LANGUAGE OverloadedStrings #-}
import Reflex.Dom

main = mainWidget $ el "div" $ text ”Hello Bob 2017"
```

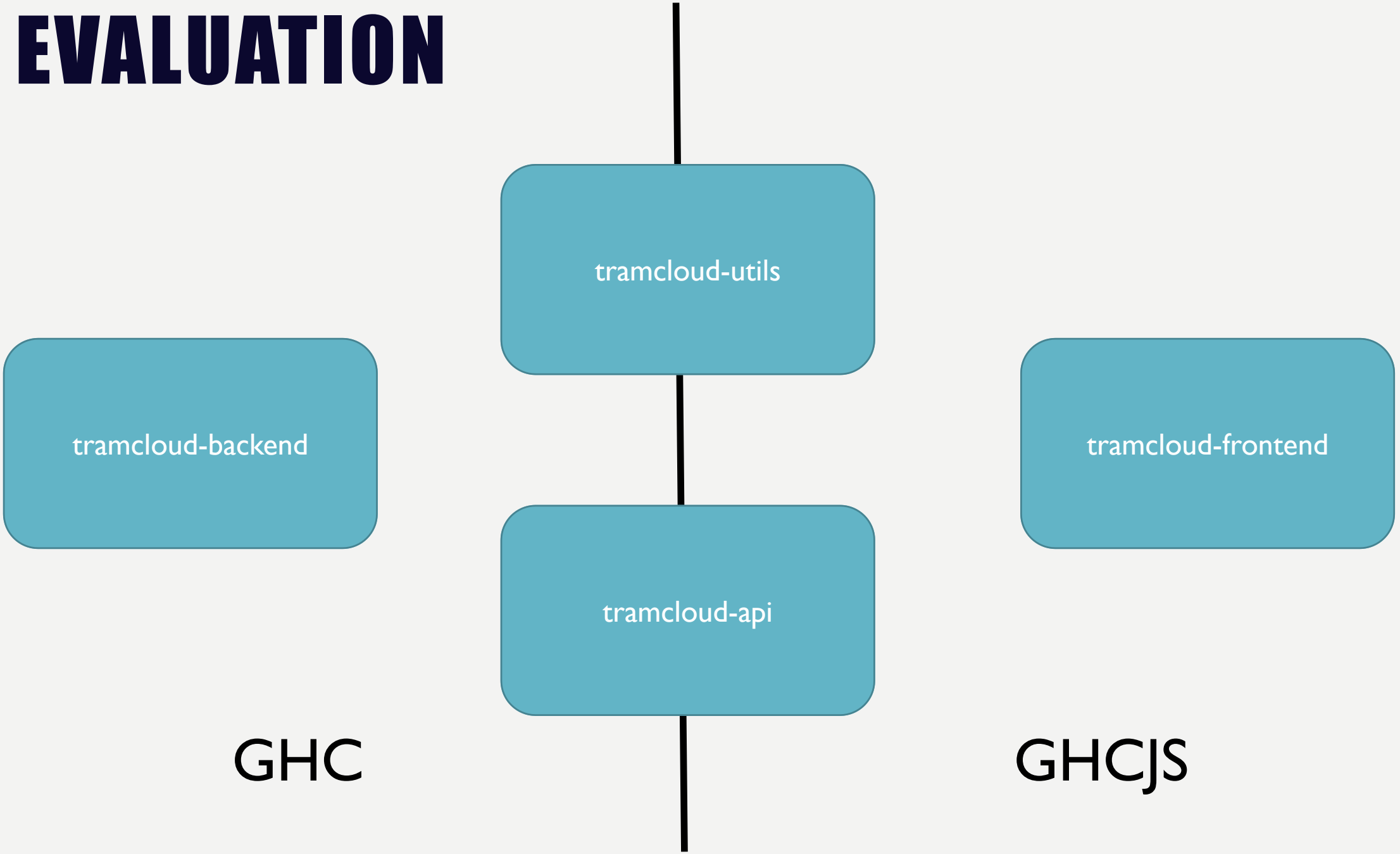
HASKELL (GHCJS)

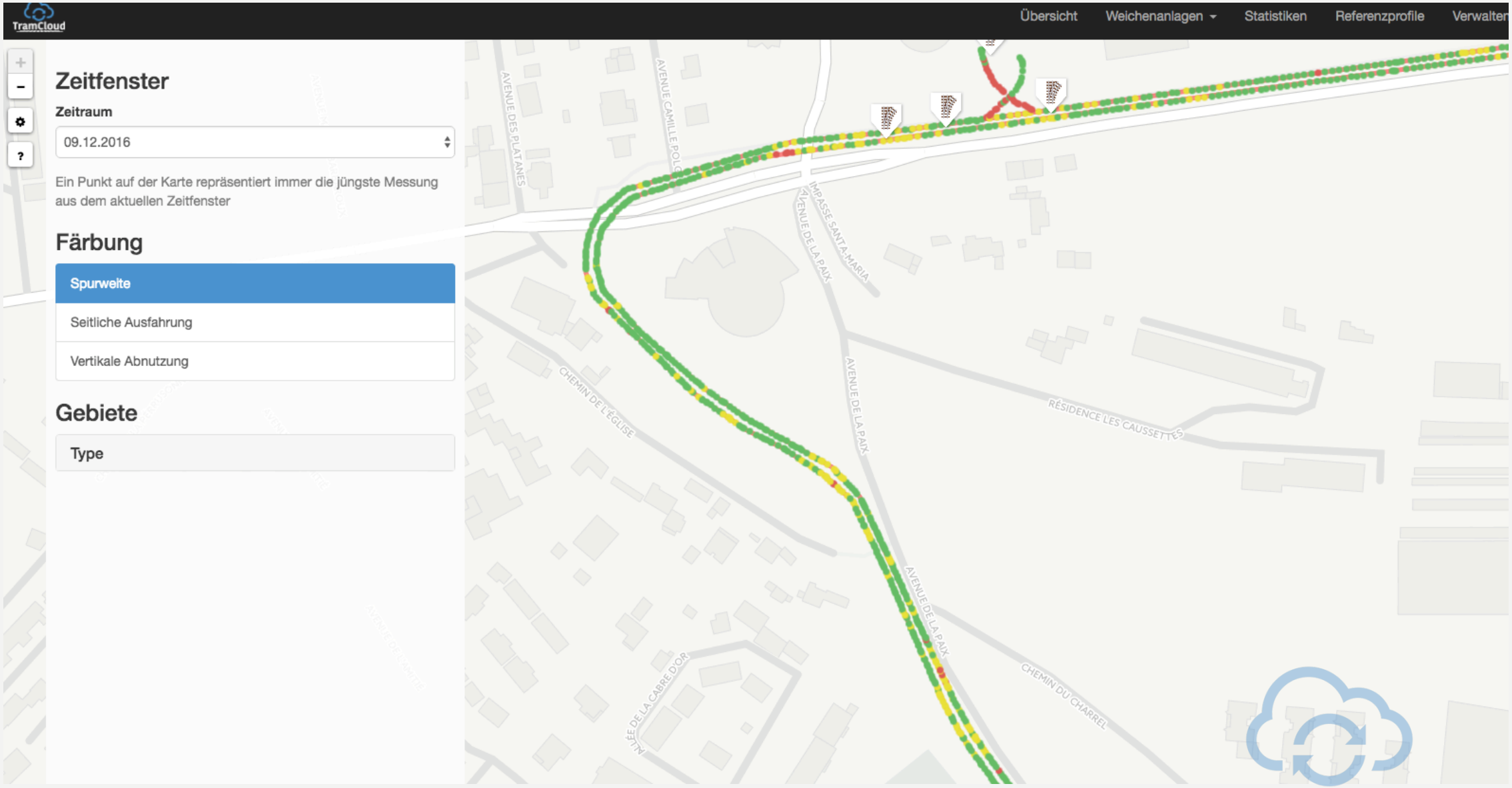
- Full Haskell in the browser
  - Runtime: Closely match semantics and execution model of GHC
  - Compiler: piggy back on GHC
- slower development, manly the creator (Luite Stegeman)
- Good ecosystem
  - Many Hackage packages
  - Wrapping existing JavaScript code is possible
- Used at Obsidian Systems (Reflex FRP), FP Complete, …

EVALUATION

- React-flux library
- Small example project
  - React, 3rd Party React Components
  - I18n
  - Code mimification, JavaScript Bundling
  - Canvas
  - Concurrency
  - IE9+ Support
    - Could not open debug console in IE8: It crashed
  - Final JS "binary" with all dependencies: 1.1MB (gzipped: 253 KB)

EVALUATION


tramcloud-utils
tramcloud-backend
tramcloud-frontend
tramcloud-api
GHC
GHCJS


TramCloud
Übersicht
Weichenanlagen
Statistiken
Referenzprofile
Verwalten
Zeitfenster
Zeitraum
09.12.2016
Ein Punkt auf der Karte repräsentiert immer die jüngste Messung aus dem aktuellen Zeitfenster
Färbung
Spurweite
Seitliche Ausfahrung
Vertikale Abnutzung
Gebiete
Type
AVENUE DES PLATANES
AVENUE CAMILLE POLO
IMPASSE SANTA-MARIA
AVENUE DE LA PAIX
CHEMIN DE L'ÉGLISE
RÉSIDENCE LES CAUSSETTES
ALLÉE DE LA CABRE D'OR
CHEMIN DU CHARREL

TramCloud

EVALUATION

- Implemented TramCloud map overview
- Very happy
  - Haskell ❤️
  - reuse existing code
- Problems
  - High iteration times (GHC is slow 😴)
  - Performance penalty when going back and forth to and from JS often
  - Runtime errors hard to debug
    - Massive amounts of JavaScript
    - Slow/Broken debugger
  - Type system does not optimally map to JavaScript's objects (basically: missing row polymorphism / anonymous records)

GHCJS

PRO	"CON"
Haskell • Static types • Immutability • Green threads	• High iteration times • Output: Massive amount of unreadable JavaScript code • Interop with JavaScript too expensive • Type system does not match perfectly • Average editor support

PURESCRIPT

```
import Test.QuickCheck

main =
    do quickCheck $ \xs ys ->
           isSorted $ merge (sort xs) (sort ys)
       quickCheck $ \xs ys ->
           xs `isSubarrayOf` merge xs ys
```

PURESCRIPT

- Looks a lot like Haskell, but has differences like row polymorphism
- Purity, strong static type system, ...
- GOAL: generate (human) readable JavaScript
- Steady development by 5-10 regular contributors, frequent releases
- Good ecosystem
  - useful PureScript packages (605, 358 documented) exist
  - easy to wrap npm packages
- Used at SlamData, Symbolian, TramCloud (hi!), ...

Explorer

← zurück zur Karte

Spurweite: 1436 mm

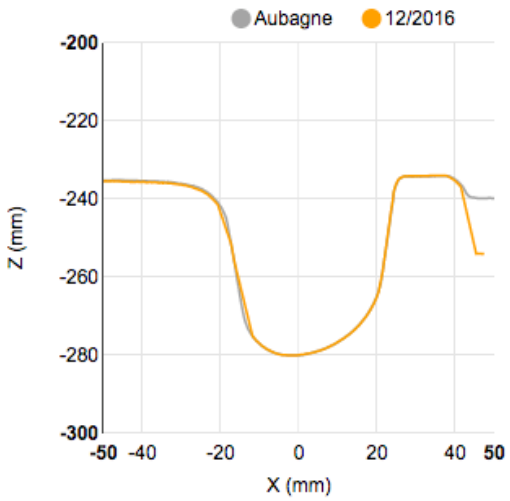
Messung vergleichen

Linke Schiene in Fahrtrichtung

Messung	Datum	seitl.	vertik.	Breite	Tiefe
167159	12/2016	0 mm	0 mm	41 mm	44 mm

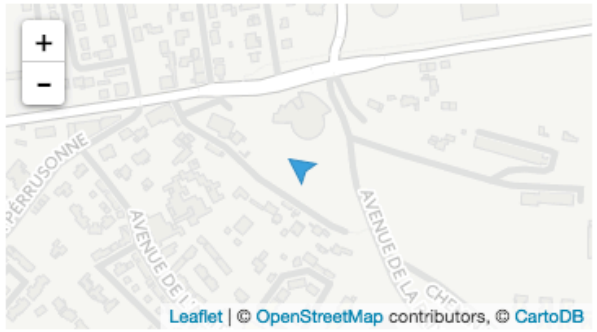
Rechte Schiene in Fahrtrichtung

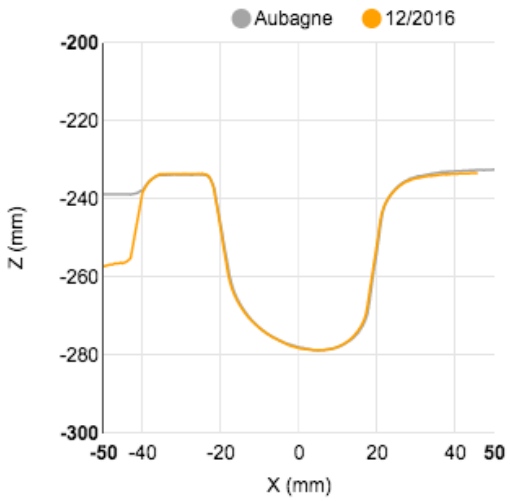
Messung	Datum	seitl.	vertik.	Breite	Tiefe
167159	12/2016	0 mm	1 mm	41 mm	45 mm

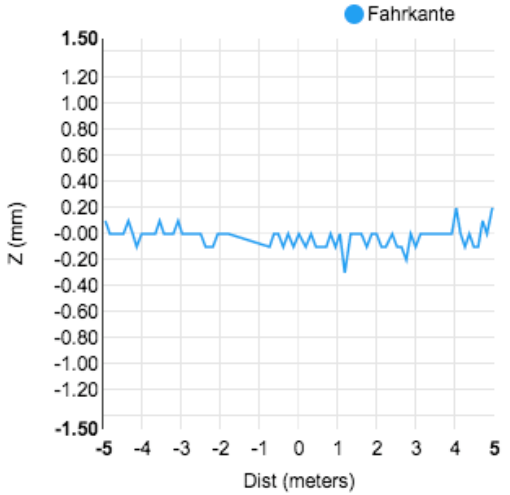


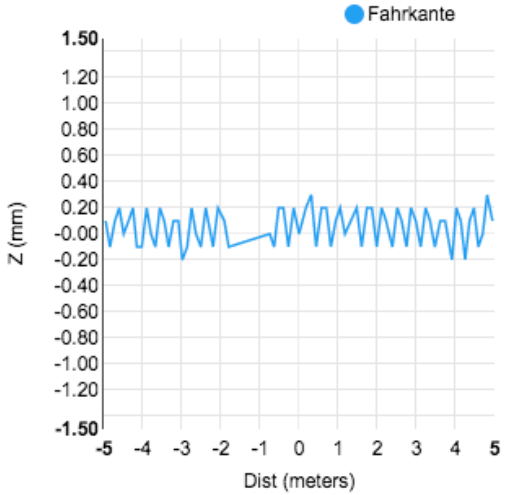
↑ in Fahrtr.↑ | ↓ gegen die Fahrtr.↓

ca. 1m in bzw. gegen die Fahrtrichtung bewegen









TramCloud

TRAMCLOUD'S PURESCRIPT STORY

- Finished porting measurement detail view, more in progress
- Built on purescript-react and many FFI wrapped npm dependencies like (d3, nvd3, spinkit, leaflet, moment, …)
- Easy to pick up with prior Haskell knowledge
- Awesome (emacs) Editor support 🚀
- Very productive language for this task
- Problems
  - Syntax more verbose than Haskell
  - Missing counterpart to TemplateHaskell(e.g. for lenses)
  - Sometimes: Notable runtime performance hits (all functions are curried)

PURESCRIPT

PRO	"CON"
• Easy to learn coming from Haskell • Great tooling (Emacs plugin) • Great ecosystem • Good integration to JavaScript world • Readable JavaScript code • Immutability • Side effect control	• Runtime performance not addressed yet • Missing features • Package management strategy not solved • No formal process for changes to language (yet?)

CONCLUSION

Use Case	TypeScript	Elm	GHCJS	PureScript
Non-FP Team, Small Project				
Non-FP Team, Big Project				
FP Team, Small Project				
FP Team, Big Project				

THANK YOU!

TWITTER: @AGRAFIX
EMAIL: MAIL@ATHIEMANN.NET