

# Gradual typing in JavaScript

Дмитрий Локтев <[dima.loktev@frontendy.com](mailto:dima.loktev@frontendy.com)>

HolyJS 2016



```
ERROR in ./index.js
Module build failed: SyntaxError: /Users/potomushto/Projects/holyjs2106/index.js: Unexpected token (6:7)
   4 | import Presentation from "./presentation";
     | ^
   5 |
>  6 | render(<Presentation/>, document.getElementById("root"));
     |         ^
   7 |
    at Parser.pp.raise (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:1378:13)
    at Parser.pp.unexpected (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:2817:8)
    at Parser.pp.parseExprAtom (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:749:12)
    at Parser.pp.parseExprSubscripts (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:504:19)
    at Parser.pp.parseMaybeUnary (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:484:19)
    at Parser.pp.parseExprOps (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:415:19)
    at Parser.pp.parseMaybeConditional (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:397:19)
    at Parser.pp.parseMaybeAssign (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:360:19)
    at Parser.pp.parseExprListItem (/Users/potomushto/Projects/holyjs2106/node_modules/babylon/index.js:1206:16)
```

```
1 function(t) {
2   function n(e) {
3     if (r(e))
4       return r(e).exports;
5     var i = r(e) = {
6       exports: {},
7       id: e,
8       loaded: !1
9     };
10    return t[e].call(i.exports, i, i.exports, n),
11    i.loaded = !0,
12    i.exports
13  }
14  var r = {};
15  return n.m = t,
16  n.c = r,
17  n.p = "/dist/",
18  n(0)
19 }((function(t, n, r) {
20   r(111),
21   function() {
22     throw new Error('Cannot find module "./index!')
23   }
24 }, function(t, n, r) {
25   var e = r(4)
26   i = r(76)
27 })
```

```
undefined is not an object (evaluating
require('NativeModules').UIManager.AndroidDraw
erLayout.Constants')
<unknown>
Browser Lays Android.android.js:13
require
require.js:249
<unknown>
react-native.js:24
require
require.js:249
<unknown>
index.ios.js:7
require
require.js:249
<unknown>
bundle.js:105
require
require.js:105
<unknown>
bundle.js:1
Dismiss (ESC) Reload JS (RR)
```

# Node.js, Chrome, iOS

# COMBO 3x

# NaN основных проблем

- Неявные преобразования типов [object Object]
- Cannot read property 'foo' of null
- Cannot set property 'foo' of null
- undefined is not a function



TJ Holowaychuk

Jul 3, 2014 · 6 min read

# Farewell Node.js

Leaving node.js land

I've been fighting with Node.js long enough in production now that I don't enjoy working with it anymore unfortunately, so at least for now this my formal farewell! And more importantly I need maintainers!

# Evolution

- REPL
- Run-time контракты
- Поддержка со стороны IDE
- Real-time test coverage
- Другой язык
- ...types?

*Strong*

*Dynamic*

*Static*

*Weak*

# Опционально и постепенно

- Strongtalk = Smalltalk + types (1993)
- Typed Racket (2008)
- TypeScript: Ответ Microsoft на большой спрос (2012)
- Flow от Facebook (2014)

# Flow vs TypeScript vs ESLint

# Окружение (1/2)

```
npm i eslint@2.9.0  
npm i eslint-config-airbnb@8.0.0  
npm i typescript@1.8.10  
npm i flow-bin@0.24.0  
node -v # v6.1.0
```

# Окружение (2/2)

```
eslint example$1.js
```

```
tsc example$1.ts
```

```
flow example$1.js
```

```
node example$1.js
```

```
1. /* @flow */
2. function getLength(x) {
3.     return x.length;
4. }
5.
6. function sum(a, b) {
7.     return a + b;
8. }
9.
10. evalAndLog(() => getLength('string'));
11. evalAndLog(
12.     () => getLength(sum('string', {}))
13. );
```

# Live

```
..      ~~~~~ ~,
8.  }
9.
10. function sum(a, b) {
11.   if (typeof b !== 'number') {
12.     return 0;
13.   }
14.   return a + b;
15. }
16.
17. evalAndLog(() => getLength('string'));
18. evalAndLog(
19.   () => getLength(sum('string', {}))
20. );
21. evalAndLog(() => getLength(3));
22. evalAndLog(() => getLength());
```

= 6

# Bonus (WAT by garybernhardt)



# Как попробовать Flow на проекте

- **Шаг 1. Без вмешательства в код проекта.**
- Шаг 2. Добавляем внешние декларации, которые видит только Flow
- Шаг 3. Flow comments в коде (всё ещё не нужен build step)
- Шаг 4. Type annotations + Babel

```
$ git clone $nlp_compromise.git  
$ flow init  
$ flow check --all
```

```
... 993 more errors  
(only 50 out of 1043  
errors displayed)
```

# Как попробовать Flow на проекте

- Шаг 1. Без вмешательства в код проекта.
- **Шаг 2. Добавляем внешние декларации, которые видит только Flow**
- Шаг 3. Flow comments в коде (всё ещё не нужен build step)
- Шаг 4. Type annotations + Babel

```
// s3example.js
import { downloadFile } from 's3client'

const res = await downloadFile('example')
```

# Как попробовать Flow на проекте

- Шаг 1. Без вмешательства в код проекта.
- Шаг 2. Добавляем внешние декларации, которые видит только Flow
- **Шаг 3. Flow comments в коде (всё ещё не нужен build step)**
- Шаг 4. Type annotations + Babel

```
/*::  
type User = {  
    name: ?string  
}  
*/
```

```
function getUser_name(user/*: User*/)  
/*: ?string*/ {  
    return user.name;  
}
```

```
/**  
 * @typedef {Object} [Type Name]  
 * @property {[Type Expression]} [Propert  
 * @property {[Type Expression]} [Propert  
 * ...  
 */
```

# Как попробовать Flow на проекте

- Шаг 1. Без вмешательства в код проекта.
- Шаг 2. Добавляем внешние декларации, которые видит только Flow
- Шаг 3. Flow comments в коде (всё ещё не нужен build step)
- **Шаг 4. Type annotations + Babel**

```
export type GreeterState = {  
  name: string;  
  partOfDay?:  
    'morning' | 'afternoon' | 'evening';  
}
```

```
function hello (action, state):  
  GreeterState {  
    return state;  
}
```

```
...
```

```
import type { GreeterState }
      from './reducers/types'

const Greeter = ({ name }: GreeterState) =
  <div>Hello, {name}!</div>

const GreeterContainer =
  connect(state => state.greeter)(Greeter)

const example =
  <GreeterContainer name='Holy.JS' />
```

# СЛОЖНОСТИ

- Крайне динамичная природа Javascript (e.g. reflection)
- ECMAScript 2020
- Разные окружения
- "Синдром" eslint-disable-next-line

```
72.
73. [options]
74. module.system=haste
75.
76. esproposal.class_static_fields=enable
77. esproposal.class_instance_fields=enable
78.
79. munge_underscores=true
80.
81. module.name_mapper='^image![a-zA-Z0-9$_-]+$' -
> 'GlobalImageStub'
82. module.name_mapper='^[./a-zA-Z0-9$_-]+\.\.
(bmp\|gif\|jpg\|jpeg\|png\|psd\|svg\|webp\|m4v\|mov
\|mp4\|mpeg\|mpg\|webm\|aac\|aiff\|caf\|m4a\|mp3\|w
av\|html\|pdf\)$' -> 'RelativeImageStub'
83.
```

```
require("image!my_logo")
```

```
86. suppress_type=$FixMe
87.
```

# Learn Flow

- Тесты в официальном репозитории  
<https://github.com/facebook/flow/>
- Готовые type definitions (node, browser)
- A Deepdive into Flow (React Europe 2016)
- [tryflow.org](http://tryflow.org)

# Links

- [Wat talk: https://www.destroyallsoftware.com/talks/wat](https://www.destroyallsoftware.com/talks/wat)
- [nuclide.io](https://nuclide.io)
- <https://github.com/DefinitelyTyped/DefinitelyTyped>
- <https://github.com/flowtype/flow-typed>
- <https://github.com/flowtype/flow-for-vscode>

# Gradual typing is a mainstream

- Ember, Vue.js, Koa используют JSDoc
- Angular2, RxJS на TypeScript
- React, особенно ReactNative - Flow

<http://potomushto.com/lab/holyjs/>

 **potomushto**

**</Deck><Questions>**

