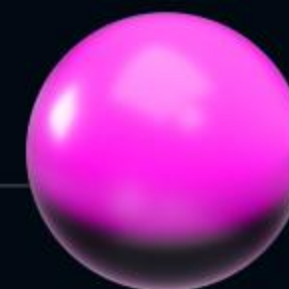


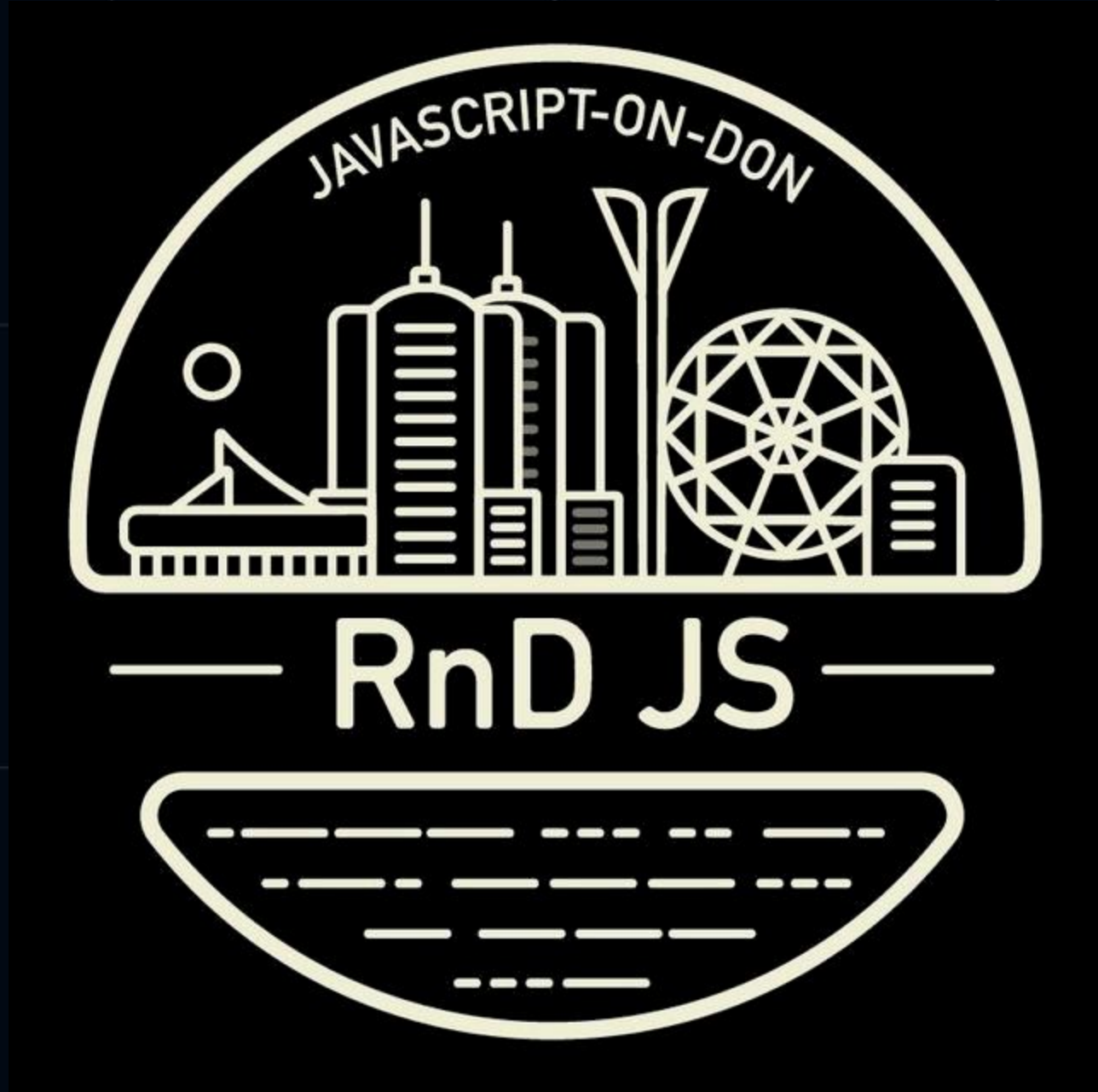
# Real DTO



**Виктор  
Вершанский**









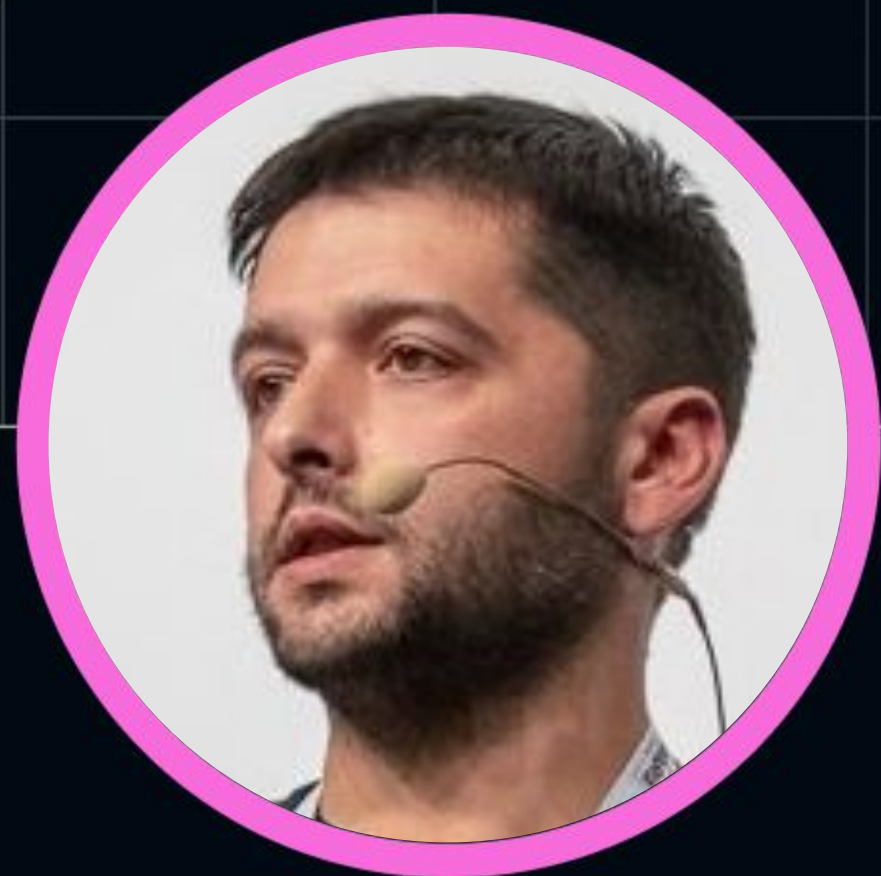
**JS**



**JS**





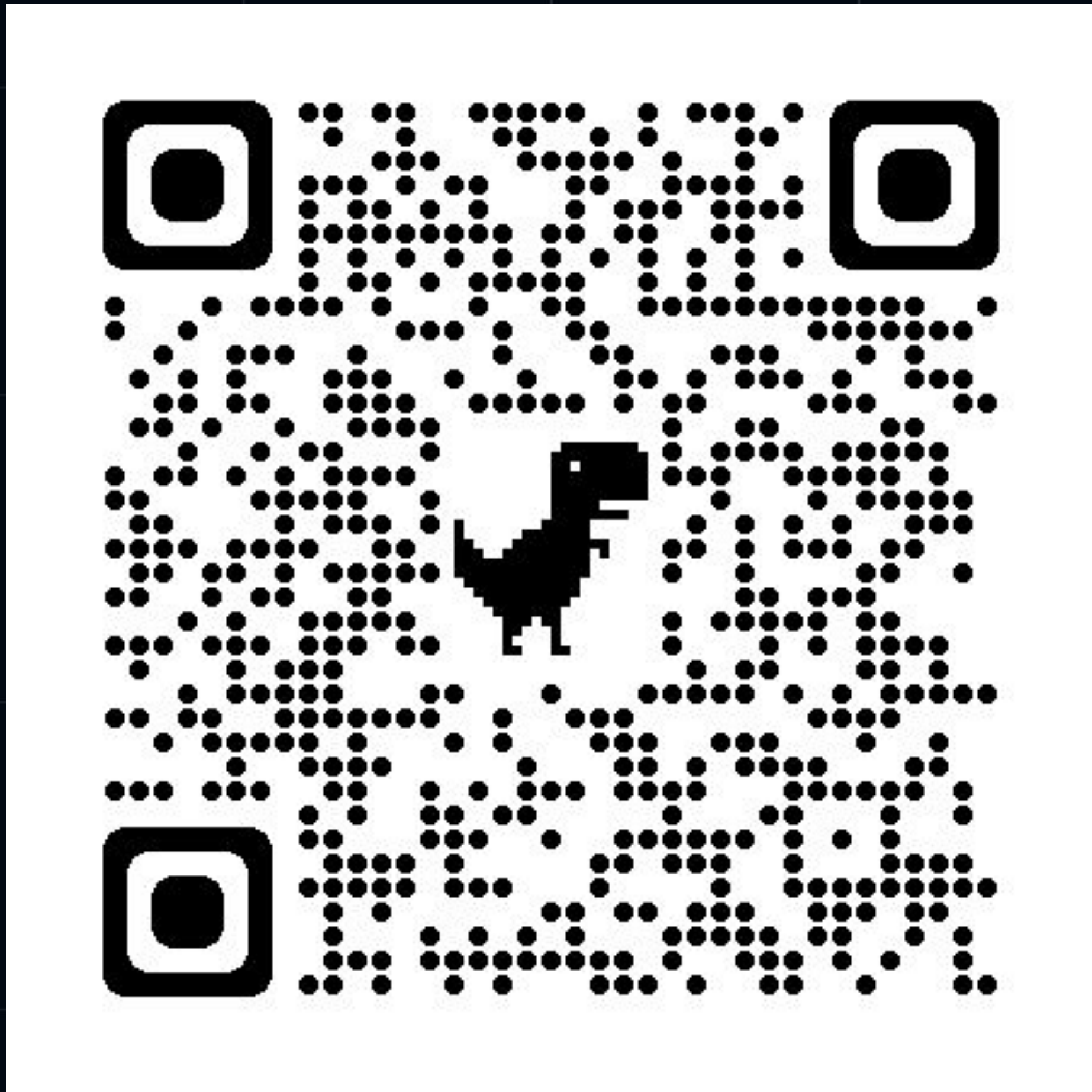


**Виктор**

 wentout

# Bio

- JS в продакшен 1999
- Back-End на JS в 2000
- Node.js с 2009
- Diagnostics Group
- BUGs Chrome & v8
- PhD in Economy of IT
- PMI PMBoK + Agile



# о чём будет идти речь

- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- как создаётся код для решения
- как развивается решение: что сделано
- практический пример использования
- что ещё можно сделать дальше





**КОНТЕКСТ**





# КОНТЕКСТ



- **КОНТЕКСТ ПОСТАНОВКИ ЗАДАЧИ**



- **КОНТЕКСТ ПОСТАНОВКИ ЗАДАЧИ**

Вопросы на собеседовании



Код с продакшена





- **КОНТЕКСТ ПОСТАНОВКИ ЗАДАЧИ**



Вопросы на собеседовании



Код с продакшена



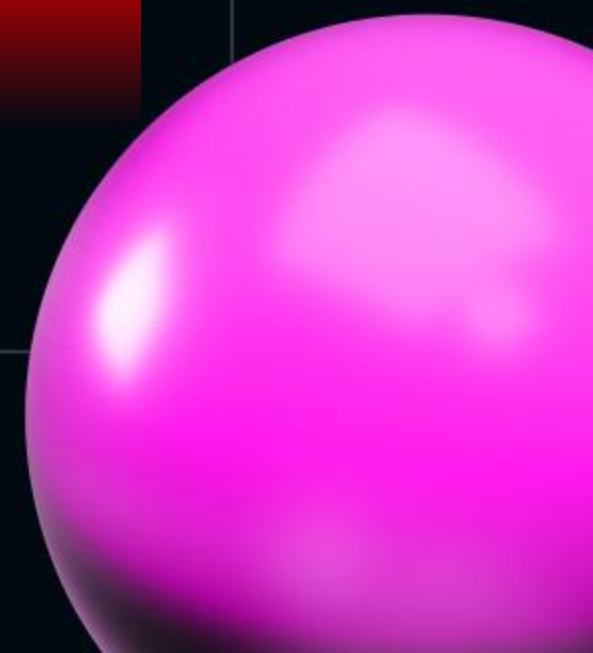




# Fata Morgana



# Fata Morgana







# Fata Morgana (mirage)

 38 languages ▾

Article [Talk](#)

Read [Edit](#) [View history](#) [Tools](#) ▾


From Wikipedia, the free encyclopedia

A **Fata Morgana** (Italian: [ˌfata moɾˈɡaːna]) is a complex form of [superior mirage](#) visible in a narrow band right above the horizon. The term *Fata Morgana* is the Italian translation of "Morgan the Fairy" ([Morgan le Fay](#) of Arthurian legend). These mirages are often seen in the Italian [Strait of Messina](#), and were described as fairy castles in the air or false land conjured by her magic.

Fata Morgana mirages significantly distort the object or objects on which they are based, often such that the object is completely unrecognizable. A Fata Morgana may be seen on land or at sea, in polar regions, or in deserts. It may involve almost any kind of distant object, including boats, islands, and the coastline. Often, a Fata Morgana changes rapidly. The mirage comprises several inverted (upside down) and erect (right-side up) images that are stacked on top of one another. Fata Morgana mirages also show alternating compressed and stretched zones.<sup>[1]</sup>

The [optical phenomenon](#) occurs because rays of light bend when they pass through air layers of different temperatures in a steep [thermal inversion](#) where an [atmospheric duct](#) has formed.<sup>[1]</sup> In calm weather, a layer of significantly warmer air may rest over colder dense air, forming an atmospheric duct that acts like a refracting [lens](#), producing a series of both inverted and erect images. A Fata Morgana requires a duct to be present; thermal inversion alone is not enough to produce this kind of mirage. While a thermal inversion often takes place without there being an atmospheric duct, an atmospheric duct cannot exist without there first being a thermal inversion.



A Fata Morgana seen over the Baltic Sea, 2016. The mirage consists of multiple upright and inverted images over the original object 









**как всё началось ...**



ukit

[Почему мы?](#)

[Цены](#)



### Сайты uKit доступны без перебоев

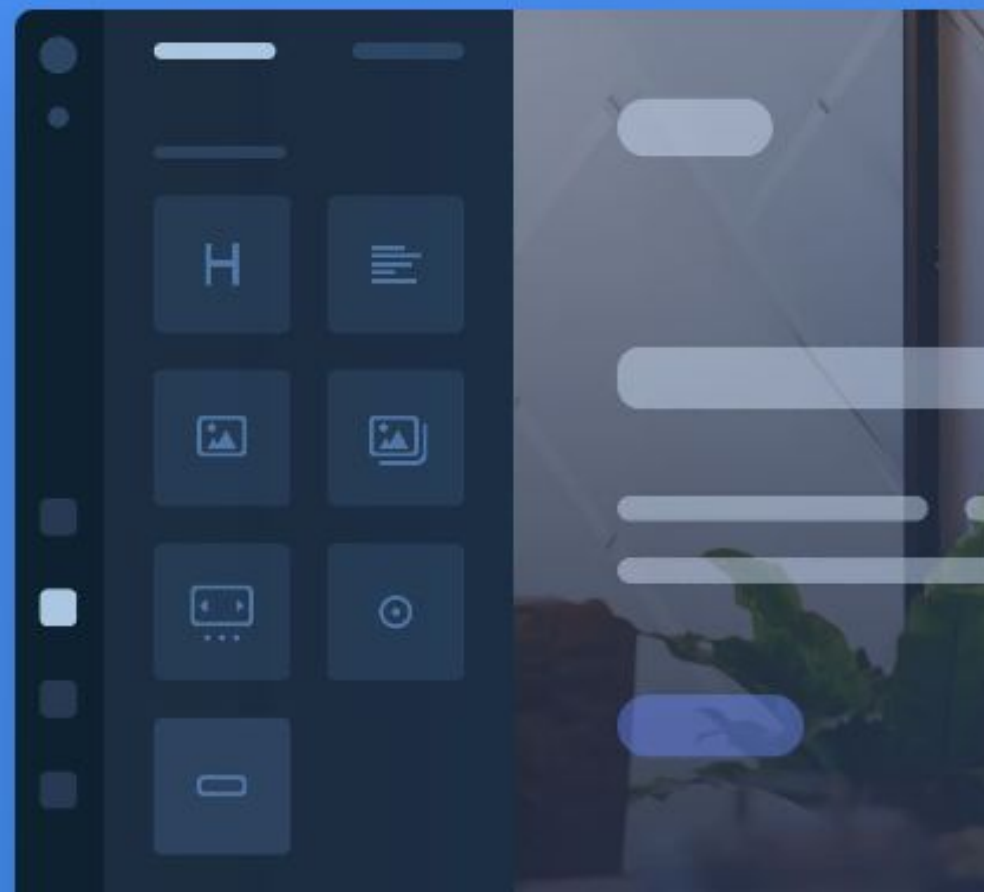
Наши серверы находятся на территории России, доступ к сайтам работает без ограничений. Конструктор uKit зарегистрирован в Едином реестре российского ПО

# Конструктор сайтов для бизнеса

Создайте сайт своими руками

[Создать сайт](#)

[▶ Смотреть видео](#)



















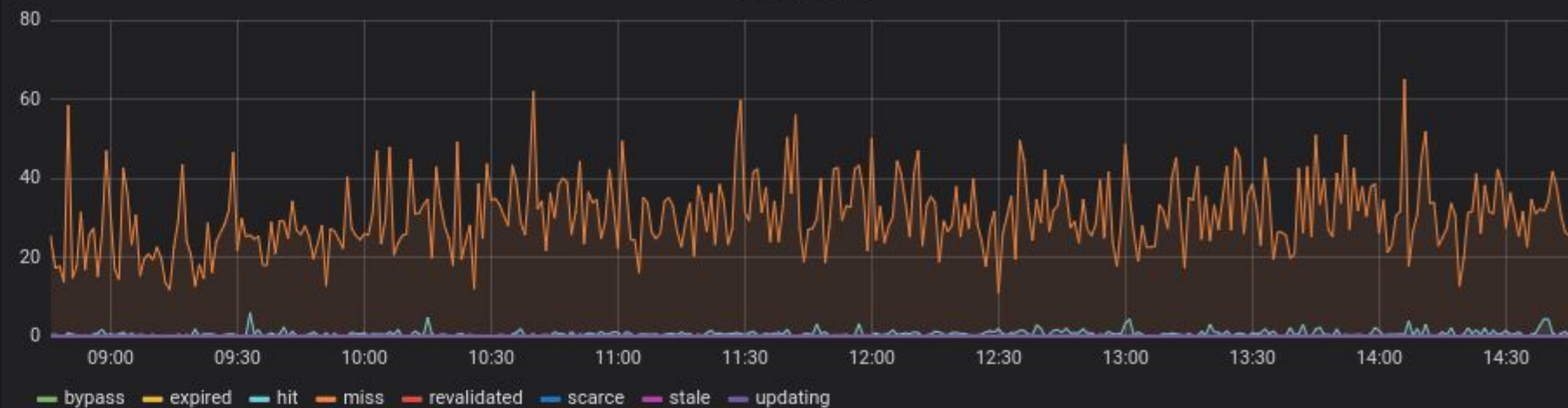




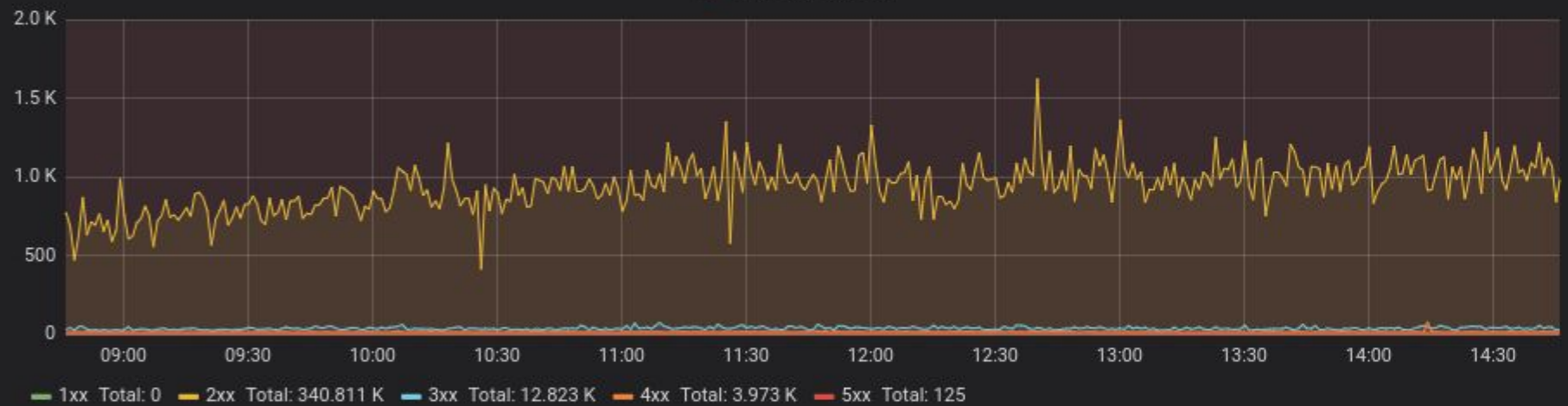
### Server Connections



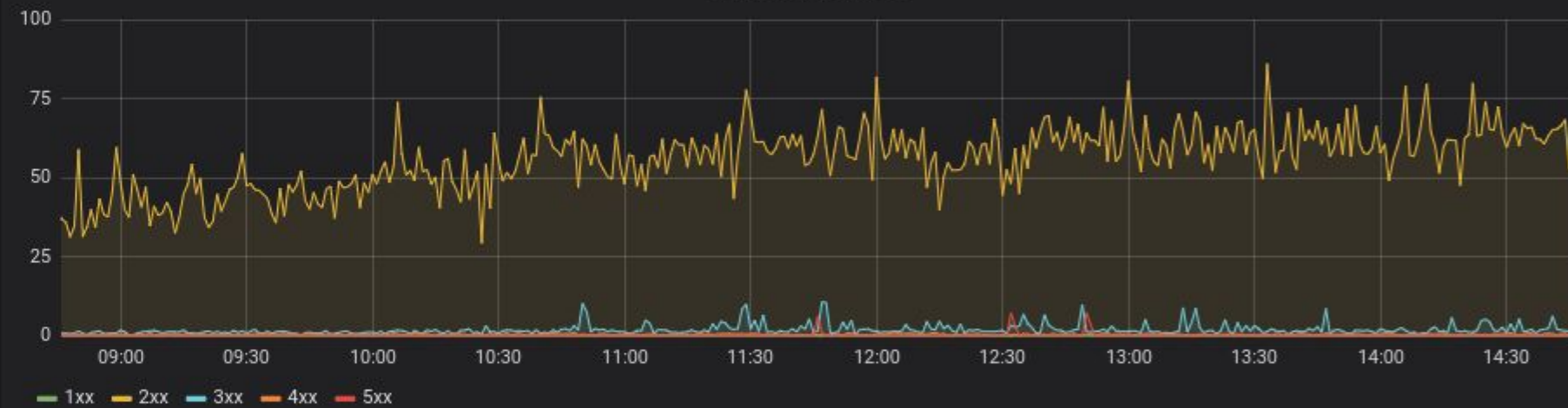
### Server Cache



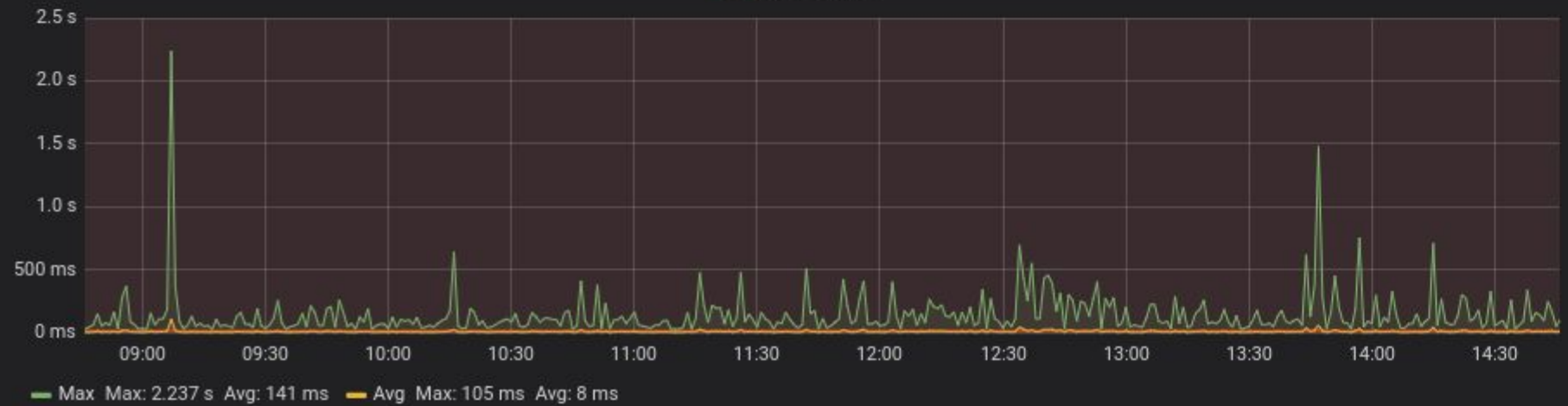
### Server Requests



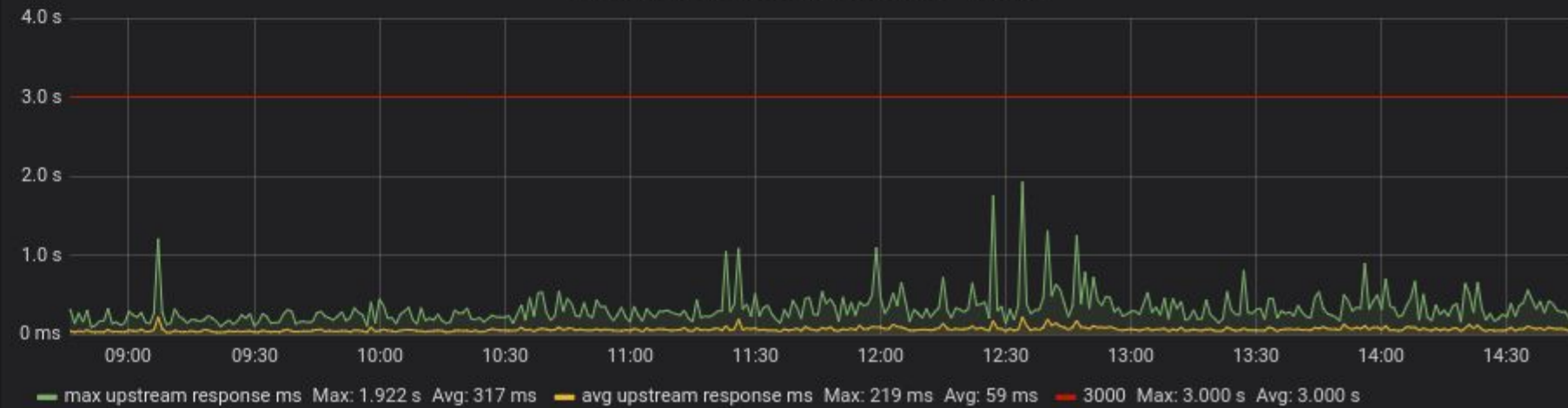
### Upstream Requests



### Request/ms



### Upstream res/ms [без конвертера и скалера]









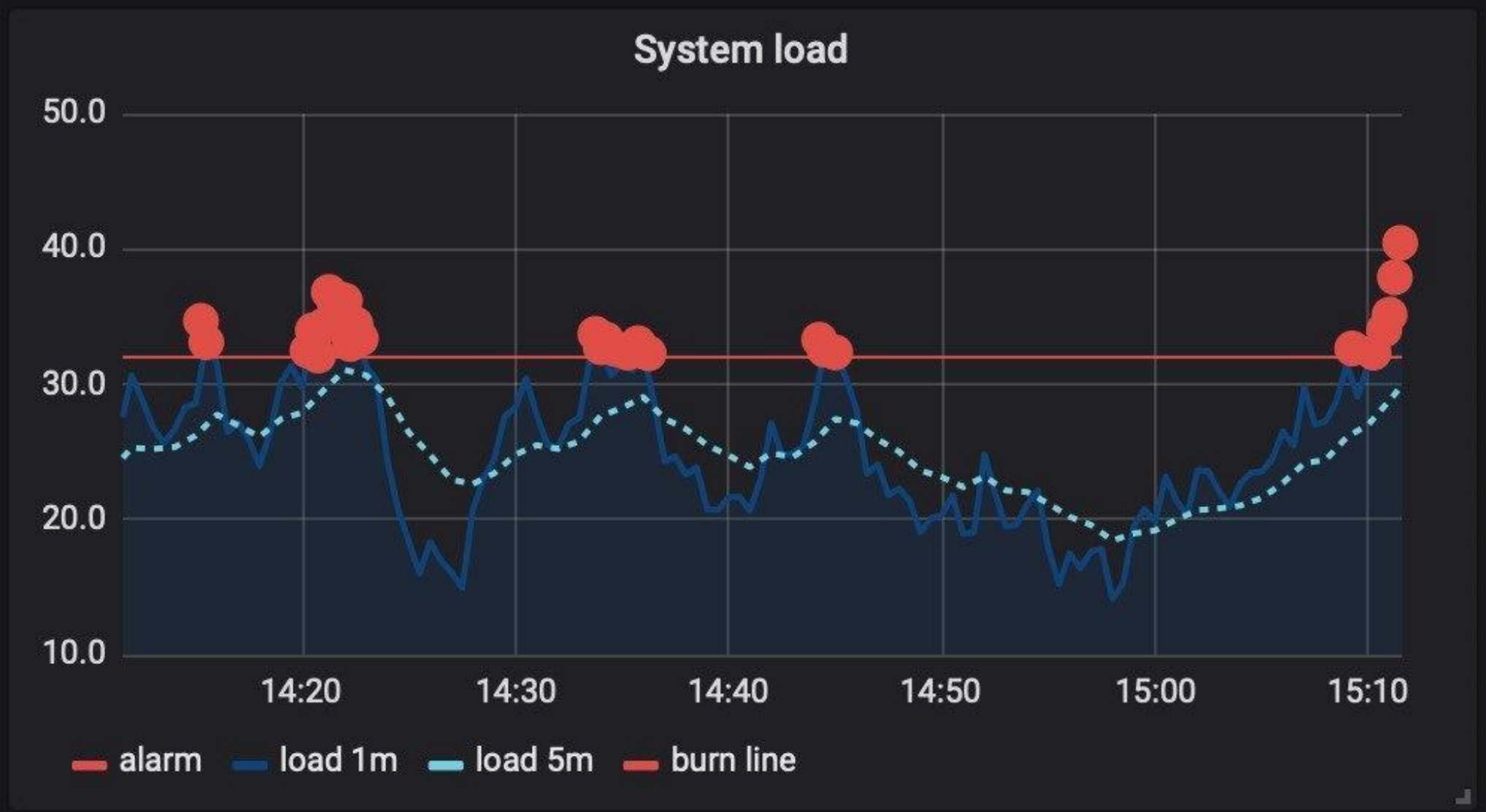
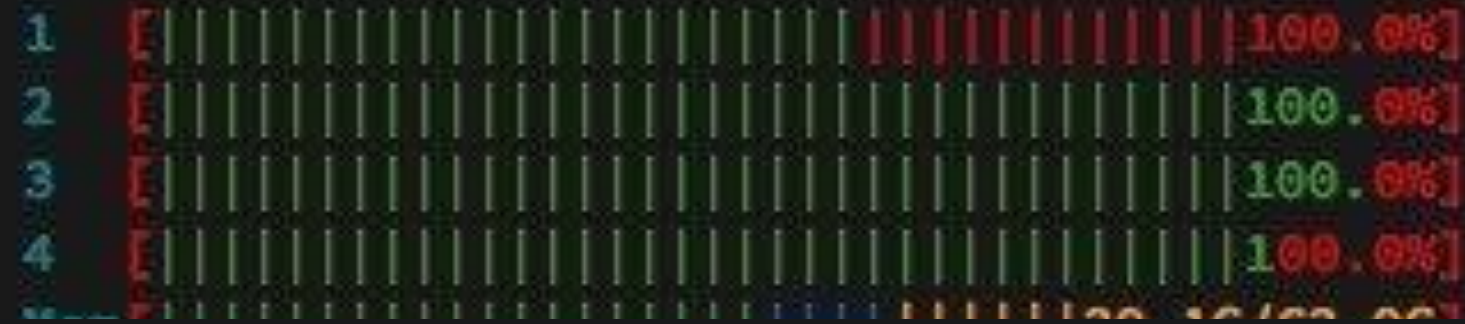


```
1 [||||| 100.0%]
2 [||||| 100.0%]
3 [||||| 100.0%]
4 [||||| 100.0%]
Mem [||||| 29.1G/62.9G]
CPU [||||| 100.0%]
```

```
5 [||||| 100.0%]
6 [||||| 100.0%]
7 [||||| 100.0%]
8 [||||| 100.0%]
Tasks: 420; 9 running
Load: 1.00 1.00 1.00
```





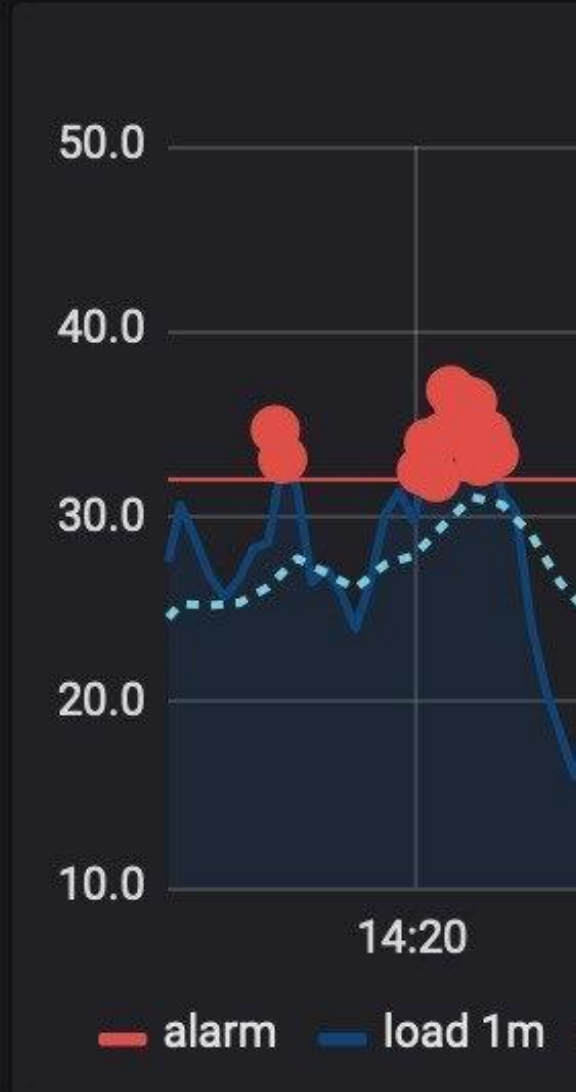


Current free memory %





1	[      100.0%]	5	[      100.0%]
2	[      100.0%]	6	[      100.0%]
3	[      100.0%]	7	[      100.0%]
4	[      100.0%]	8	[      100.0%]



```

Message from syslogd@ul2 at Jan 19 12:19:51 ...
kernel:[30582052.591040] NMI watchdog: BUG: soft lockup - CPU#10 stuck for 22s! [pidof:8724]

Message from syslogd@ul2 at Jan 19 12:19:54 ...
kernel:[30582055.947062] NMI watchdog: BUG: soft lockup - CPU#2 stuck for 23s! [pidof:9048]

Message from syslogd@ul2 at Jan 19 12:20:03 ...
kernel:[30582064.751125] NMI watchdog: BUG: soft lockup - CPU#12 stuck for 23s! [pidof:7398]

Message from syslogd@ul2 at Jan 19 12:20:10 ...
kernel:[30582071.787175] NMI watchdog: BUG: soft lockup - CPU#0 stuck for 22s! [pidof:8730]

Message from syslogd@ul2 at Jan 19 12:20:10 ...
kernel:[30582071.867174] NMI watchdog: BUG: soft lockup - CPU#1 stuck for 22s! [pidof:6775]

Message from syslogd@ul2 at Jan 19 12:20:11 ...
kernel:[30582072.191175] NMI watchdog: BUG: soft lockup - CPU#5 stuck for 23s! [pidof:7869]

Message from syslogd@ul2 at Jan 19 12:20:11 ...
kernel:[30582072.351177] NMI watchdog: BUG: soft lockup - CPU#7 stuck for 22s! [pidof:7465]

Message from syslogd@ul2 at Jan 19 12:20:11 ...
kernel:[30582072.831181] NMI watchdog: BUG: soft lockup - CPU#13 stuck for 22s! [pidof:7892]

Message from syslogd@ul2 at Jan 19 12:20:11 ...
kernel:[30582072.911180] NMI watchdog: BUG: soft lockup - CPU#14 stuck for 22s! [zabbix_agentd:6350]

```



1 [|||||  
2 [|||||  
3 [|||||  
4 [|||||

50.0  
40.0  
30.0  
20.0  
10.0  
— alarm

```
2018-10-11 13:00 +03:00: e 10.11 13:00:52 routes/site.js:620:14 error on send flash-promo mail
err:
TypeError:
  i18n/tr/mails/billing/flash-promo.html:43
    41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
erdana, sans-serif;" >
    42|         <tr>
    >> 43|             <%- putSocials(language, false, true) %>
    44|         </tr>
    45|     </table>
    46| </td>

Cannot read property 'forEach' of undefined
path: i18n/tr/mails/billing/flash-promo.html
stack:
  41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
erdana, sans-serif;" >
  42|         <tr>
  >> 43|             <%- putSocials(language, false, true) %>
  44|         </tr>
  45|     </table>
  46| </td>

Cannot read property 'forEach' of undefined
Object.putSocials (/home/www/ulight/lib/mailHelpers.js:179:18)
eval (eval at exports.compile (/home/www/ulight/lib/ejs/ejs.js:243:14), <anonymous>:30:2583)
eval (eval at exports.compile (/home/www/ulight/lib/ejs/ejs.js:243:14), <anonymous>:30:39381)
/home/www/ulight/lib/ejs/ejs.js:256:15
Object.exports.render (/home/www/ulight/lib/ejs/ejs.js:294:13)
readFile.cb.ok (/home/www/ulight/lib/h.js:1011:21)
/home/www/ulight/lib/ok.js:12:10
fs.readFile.cb.ok (/home/www/ulight/lib/h.js:908:5)
/home/www/ulight/lib/ok.js:12:10
/home/www/ulight/node_modules/graceful-fs/graceful-fs.js:78:16
FSReqWrap.readFileAfterClose [as oncomplete] (fs.js:511:3)
siteId: [ObjectID 5bbf1ee7e49e66357759b86b ]
userId: [ObjectID 5bbf1dbfddd387a3cacc412 ]
```

ul1

```
||||| 100.0%
||||| 100.0%
||||| 100.0%
||||| 100.0%

[pidof:8724]
merge branch 'ro112'
[Thu Jan 18 13:32:00
[pidof:9048]
[pidof:7398]
01-19 12:01:07: App
id [_old_118] and pi
[pidof:8730] signal [S
rred between 2018-01-
[pidof:6775]
[pidof:7869]
[pidof:7465]
01-19 12:18:01: App
with id [121] and pid
[pidof:7892] INT]
rred between 2018-01-
[zabbix_agentd:6350]
```





**формулировка**

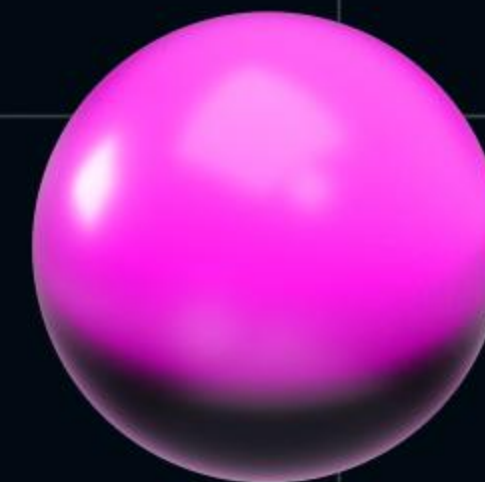


# формулировка





- КОНТЕКСТ ПОСТАНОВКИ ЗАДАЧИ
- **формулировка проблематики**







```
13|ulcraft | 2018-10-11 13:00 +03:00: e 10.11 13:00:52 routes/site.js:620:14 error on send flash-promo mail
13|ulcraft |   err:
13|ulcraft |     | TypeError:
13|ulcraft |     |   | i18n/tr/mails/billing/flash-promo.html:43
13|ulcraft |     |   | 41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
erdana, sans-serif;" >
13|ulcraft |     |   | 42|         <tr>
13|ulcraft |     |   | >> 43|                                     <%- putSocials(language, false, true) %>
13|ulcraft |     |   | 44|         </tr>
13|ulcraft |     |   | 45|     </table>
13|ulcraft |     |   | 46|     </td>
13|ulcraft |     |   | Cannot read property 'forEach' of undefined
13|ulcraft |     |   | path: i18n/tr/mails/billing/flash-promo.html
13|ulcraft |     |   | stack:
13|ulcraft |     |   | 41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
erdana, sans-serif;" >
13|ulcraft |     |   | 42|         <tr>
13|ulcraft |     |   | >> 43|                                     <%- putSocials(language, false, true) %>
13|ulcraft |     |   | 44|         </tr>
13|ulcraft |     |   | 45|     </table>
13|ulcraft |     |   | 46|     </td>
13|ulcraft |     |   | Cannot read property 'forEach' of undefined
13|ulcraft |     |   | Object.putSocials (/home/www/ulight/lib/mailHelpers.js:179:18)
13|ulcraft |     |   | eval (eval at exports.compile (/home/www/ulight/lib/ejs/ejs.js:243:14), <anonymous>:30:2583)
13|ulcraft |     |   | eval (eval at exports.compile (/home/www/ulight/lib/ejs/ejs.js:243:14), <anonymous>:30:39381)
13|ulcraft |     |   | /home/www/ulight/lib/ejs/ejs.js:256:15
13|ulcraft |     |   | Object.exports.render (/home/www/ulight/lib/ejs/ejs.js:294:13)
13|ulcraft |     |   | readFile.cb.ok (/home/www/ulight/lib/h.js:1011:21)
13|ulcraft |     |   | /home/www/ulight/lib/ok.js:12:10
13|ulcraft |     |   | fs.readFile.cb.ok (/home/www/ulight/lib/h.js:908:5)
13|ulcraft |     |   | /home/www/ulight/lib/ok.js:12:10
13|ulcraft |     |   | /home/www/ulight/node_modules/graceful-fs/graceful-fs.js:78:16
13|ulcraft |     |   | FSReqWrap.readFileAfterClose [as oncomplete] (fs.js:511:3)
13|ulcraft |     |   | siteId: [ObjectID 5bbf1ee7e49e66357759b86b ]
13|ulcraft |     |   | userId: [ObjectID 5bbf1dbfddd387a3cacc412 ]
```

ul1









**ИЗЮМА  
ТЯПНУЛИ**

```
13:00 +03:00: e 10.11 13:00:52 routes/site.js:
err:
TypeError:
    i18n/tr/mails/billing/flash-promo.html:43
    41| <table class="net2" align="right" style="b
erdana, sans-serif;" >
    42|         <tr>
    >> 43|                                     <%- pu
    44|         </tr>
    45|     </table>
    46| </td>
Cannot read property 'forEach' of undefined
```



**А ДАВАЙТЕ  
ЧТО-НИБУДЬ**

**ВНЕДРИМ**

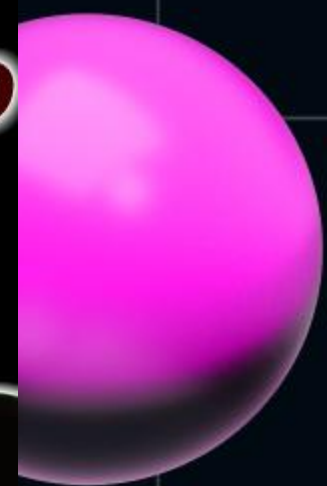


**ЭТО БЫЛО  
НЕВОЗМОЖНО**

**ЗАПЛАНИРОВАТЬ**

```
pers.js:179:18)
t/lib/ejs/ejs.js:243:14), <anonymous>:30:2583)
t/lib/ejs/ejs.js:243:14), <anonymous>:30:39381)
s/ejs.js:294:13)
(21)
08:5)
eful-fs.js:78:16
fs.js:511:3)
```

u/l1







**ХОТЯ ...**





**ХОТЯ ... если немного ...**









```
1 [||||| 100.0%]
```

```
13:32:06 93cbb5
```

```
App and pi gnal [9
```

```
918-01-
```

```
App and pid ENT]
```

```
red between 2018-01-
```

```
[zabbix_agentd:6350]
```

```
2018-10-11 13:00 +03:00: e 10.11 13:00:52 routes/site.js:620:14 error on send flash-promo mail
```

```
13:32:06 93cbb5
```

```
App and pi gnal [9
```

```
918-01-
```

```
App and pid ENT]
```

```
red between 2018-01-
```

```
[zabbix_agentd:6350]
```

# HolyJS

// Москва 2016

---

## Thomas Watson

Opbeat

Debugging Node.js  
Performance Issues  
in Production







1  
2  
3  
4

50

40

30

20

10

0.0%]  
0.0%]  
0.0%]  
0.0%]

24]

anch 'ro112

18 13:32:0

8] 7943cbb5

98]

1:07; App

18] and pi

0] signal [S

en 2018-01-

5]

9]

5]

01; App

] and pid

92] INT]

en 2018-01-

gentd:6350



# HolyJS // Москва 2016

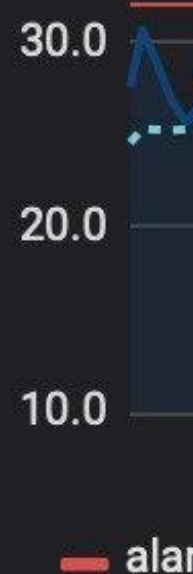
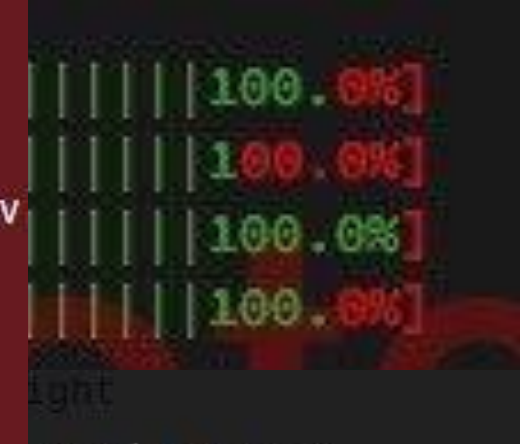
Thomas Watson

Opbeat

Debugging Node.js  
Performance Issues  
in Production



```
20:14 error on send flash-promo mail  
  
border-spacing:0;color:#3b7694;font-family:V  
  
Socials(language, false, true) %>
```



```
45|         </table>  
46|     </td>  
  
Cannot read property 'forEach' of undefined  
Object.putSocials (/home/www/ulight/lib/mailHelper  
eval (eval at exports.compile (/home/www/ulight/li  
eval (eval at exports.compile (/home/www/ulight/li  
/home/www/ulight/lib/ejs/ejs.js:256:15  
Object.exports.render (/home/www/ulight/lib/ejs/ej  
readFile.cb.ok (/home/www/ulight/lib/h.js:1011:21)  
/home/www/ulight/lib/ok.js:12:10  
fs.readFile.cb.ok (/home/www/ulight/lib/h.js:908:5  
/home/www/ulight/lib/ok.js:12:10  
/home/www/ulight/node_modules/graceful-fs/graceful  
FSReqWrap.readFileAfterClose [as oncomplete] (fs.j  
siteId: [ObjectID 5bbf1ee7e49e66357759b86b ]  
userId: [ObjectID 5bbf1dbfddd387a3cacc412 ]  
reqId: [50302072.911100] req watchdog. boo
```



BoF-CE  
BoF-SES



















**HolyJS** // Москва 2016

**Thomas Watson**

Opbeat

Debugging Node.js  
Performance Issues  
in Production



question about `async_hooks` 🐼



**system**

Hi Tomas! Sorry for interrupting! Nice to meet you! My name is Victor, I'm from Russia, and I c



**Thomas Watson**

to me ▾

If I understand your proposal correctly, you are advocating for a 5th hook to be added to Async



```
1 [||||] i@ulcraft: | 2018-10-11 13:00 +03:00: e 10.11 13:00:52 routes/site.js:620:14 error on send flash-promo mail
2 [||||] i@ulcraft: | | err:
3 [||||] i@ulcraft: | | | TypeError:
  | | | i18n/tr/mails/billing/flash-promo.html:43
  | | | 41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
    erdana, sans-serif;" >
```

```
|||| 100.0%]
|||| 100.0%]
|||| 100.0%]
```

# question about async\_hooks



system

Hi Tomas! Sorry for

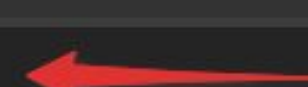


Thomas Watson

to me

If I understand your

```
context.js x
```



```
const eid = hooks.eid;
const tid = hooks.tid;
```

```
eid = 1
tid = 0
```

```
debugger;
```

```
alar i@ulcraft: | /home/
i@ulcraft: | FSReq
i@ulcraft: | siteId: [
i@ulcraft: | userId: [
```




```
1 | 13:00:52 routes/site.js:629:14 error on send flash-promo mail
2 | | err:
3 | |   TypeError:
  | |     i18n/tr/mails/billing/flash-promo.html:43
  | |       41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
    | |         erdana, sans-serif;" >
```



question about async\_hooks ➤

 **system**  
Hi Tomas! Sorry for

 **Thomas Watson**  
to me ▾  
If I understand your

 context.js ✕

0

```
const eid = hooks.eid;
const tid = hooks.tid;
```

```
eid = 1
tid = 0
```

debugger;

```
alar | /home/
      | FSReq
      | siteId: [
      | userId: [
```




```
1 | 13|ulcraft: | 2018-10-11 13:00 +03:00: e 10.11 13:00:52 routes/site.js:620:14 error on send flash-promo mail
2 | 13|ulcraft: | | err:
3 | 13|ulcraft: | | | TypeError:
  | 13|ulcraft: | | | | i18n/tr/mails/billing/flash-promo.html:43
  | 13|ulcraft: | | | | 41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
    | 13|ulcraft: | | | | erdana, sans-serif;" >
```



question about async\_hooks ➤

 **system**  
Hi Tomas! Sorry for

 **Thomas Watson**  
to me ▾  
If I understand your

 context.js ✕

0

```
const eid = hooks.eid;
const tid = hooks.tid;
```

```
eid = 1
tid = 0
```

debugger;

```
alar 13|ulcraft: | | /home/
13|ulcraft: | | FSReq
13|ulcraft: | | siteId: [
13|ulcraft: | | userId: [
  | 13|ulcraft: | | |
```







```
1 | 2018-10-11 13:00 +03:00: e 10.11 13:00:52 routes/site.js:620:14 error on send flash-promo mail
2 | | err:
3 | | | TypeError:
  | | | | i18n/tr/mails/billing/flash-promo.html:43
  | | | | 41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
    | | | | erdana, sans-serif;" >
```

# question about async\_hooks



nodejs / diagnostics

Watch 108 Star 281 Fork 54

Code Issues 53 Pull requests 4 Projects 0 Wiki Insights

## Proposal of Polling~Queue: Sync Hook Problem #249

Edit New issue

Closed wentout



wentout

Hi!

First of all  
Really nice

And, I'm very  
by myself

Sync inst

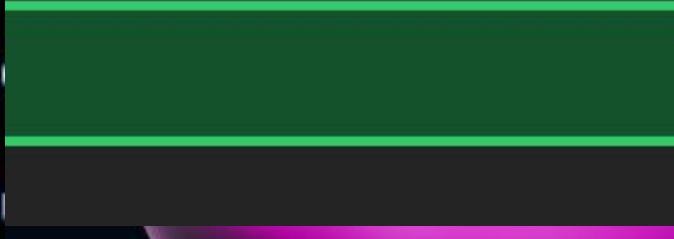
If you can

```
Error: Callback was already called.
  at /home/www/ulight/node_modules/async/lib/async.js:43:36
  at /home/www/ulight/node_modules/async/lib/async.js:694:17
  at /home/www/ulight/node_modules/async/lib/async.js:173:37
  at /home/www/ulight/node_modules/mongoose/lib/model.js:4506:16
  at model.$_save.error (/home/www/ulight/node_modules/mongoose/lib/model.js:4506:16)
  at /home/www/ulight/node_modules/kareem/index.js:315:21
  at next (/home/www/ulight/node_modules/kareem/index.js:209:27)
  at /home/www/ulight/node_modules/kareem/index.js:182:9
  at process.nextTick (/home/www/ulight/node_modules/kareem/index.js:452:38)
  at args.(anonymous function) (/home/www/.nvm/versions/node/v10.13.0/lib/node_modules/
  at process._tickCallback (internal/process/next_tick.js:61:11)
```

```
2018-11-14 13:41 +03:00: e 11.14 13:41:26 service/queue.js:69:8 QueueService |
```

```
| error:
```

= 1  
= 0





```
1 | 2018-10-11 13:00 +03:00: e 10.11 13:00:52 routes/site.js:620:14 error on send flash-promo mail
2 | | err:
3 | | | TypeError:
  | | | | i18n/tr/mails/billing/flash-promo.html:43
  | | | | 41| <table class="net2" align="right" style="border-spacing:0;color:#3b7694;font-family:V
    | | | | erdana, sans-serif;" >
```

question about async\_hooks ➡



nodejs / diagnostics

Code Issues 53 Pull requests 4

### Proposal of Polling~Queue

Closed wentout

wentout

Hi!

First of all

Really nice

And, I'm v

by myself

Sync inst

If you can



Fork 54

New issue

```
Error: Callback was
  at /home/www/uli
  at /home/www/uli
  at /home/www/uli
  at /home/www/uli
  at model.$__save
  at /home/www/uli
  at next (/home/w
  at /home/www/uli
  at process.nextT
  at args.(anonymo
  at process._tick
```

```
2018-11-14 13:41 +03:00: e 11.14 13:41:26 service/queue.js:69:8 QueueService |
| error:
```

```
6:16
oose/lib/model.js
YouTube
:27) com/channel/U
index.js:452:38)
/v10.13.0/lib/nod
:11)
```

= 1

= 0





# context-dive

## Dive to async code with Context (v2)

Using this module you are able to achieve the following functionality:

1. CLS/TLS : Continuation/Thread Local Storage ([wikipedia link](#)).
2. Performance measuring, based on the top of **Dived Context** we are running in.
3. Meaningfull UncaughtException and UnhandledRejection extras, based on CLS.

... also there is plan to develop bit more rich functionality ...

So this module allows you to wrap some execution context context via [async\\_hooks](#). The main Idea is the wrapped **execution context** must be a function, cause otherwise we will be unable to handle everything through callbacks. Core concept is about that if we have some attributes of wrapped function as callbacks, so we will wrap them too, and therefore we will be able to track the context back through that callbacks too.



```

      I
e Error:
  at process.uncaughtExceptionHandler (/home/went/_dev/context-dive.com/holy/mods/errors.js:6:6)
  at process.emit (events.js:187:15)
  at process.EventEmitter.emit (domain.js:441:20)
  at process._fatalException (internal/bootstrap/node.js:622:27)
ReferenceError: b is not defined
  at module.exports (/home/went/_dev/context-dive.com/holy/routes/fail.js:13:13)
  at userModel.findOne (/home/went/_dev/context-dive.com/holy/routes/simple.js:30:21)
  at /home/went/_dev/context-dive.com/holy/node_modules/mongoose/lib/model.js:4529:16
  at process.nextTick (/home/went/_dev/context-dive.com/holy/node_modules/mongoose/lib/query.js:2584:28)
  at process.internalTickCallback (internal/process/next_tick.js:70:11)
HTTP Server info : socket timeout reached /
HTTP Server info : GET /demo/find/782cab23cec52a9b9207176c09deb0e395cc63d1eaa55af4 500 5006
ms
5881 repl > printFile('routes/start_a')
{ Error: Cannot find module '../routes/start_a'
  at Function.Module._resolveFilename (internal/modules/cjs/loader.js:587:15)
  at Function.resolve (internal/modules/cjs/helpers.js:32:19)
  at printFile (/home/went/_dev/context-dive.com/holy/mods/repl.js:180:31) code: 'MODULE_NOT_FOUND' }
5881 repl > printFile('routes/start_a')
```







Доклады

Спикеры

Партнеры

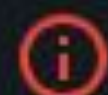
О нас

Архив

# HolyJS

2019 Piter

## Конференция для разработчиков



Мероприятие завершилось

**24–25 мая**

САНКТ-ПЕТЕРБУРГ







The Node.js logo, consisting of the letters 'n', 'o', 'd', and 'e' in a stylized, blocky font. The 'o' is a 3D orange cube, and the 'e' has a small orange hexagon inside its loop.

# The NodeConference for All JS Developers

NODE CONFERENCE NL

I amsterdam

7TH JUNE 2019, THEATER DE MEERVAART, AMSTERDAM



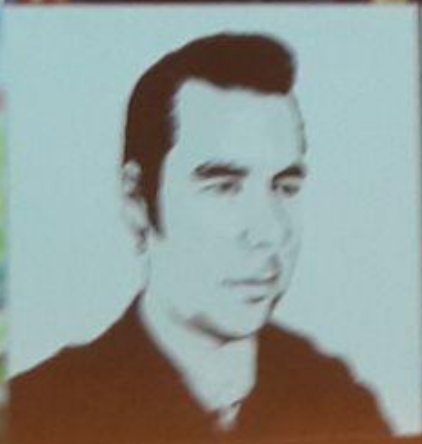


**NodeConf NL**  
[nodeconference.nl](http://nodeconference.nl)





# Speakers





# After Party!

Join us up from 19:30 at Kanarie Club

Bellamyplein 51, 1053 AT Amsterdam

Direct Tram Line 17 towards Central Station  
Get off at Stop: Ten Katestraat (1 min. walking)













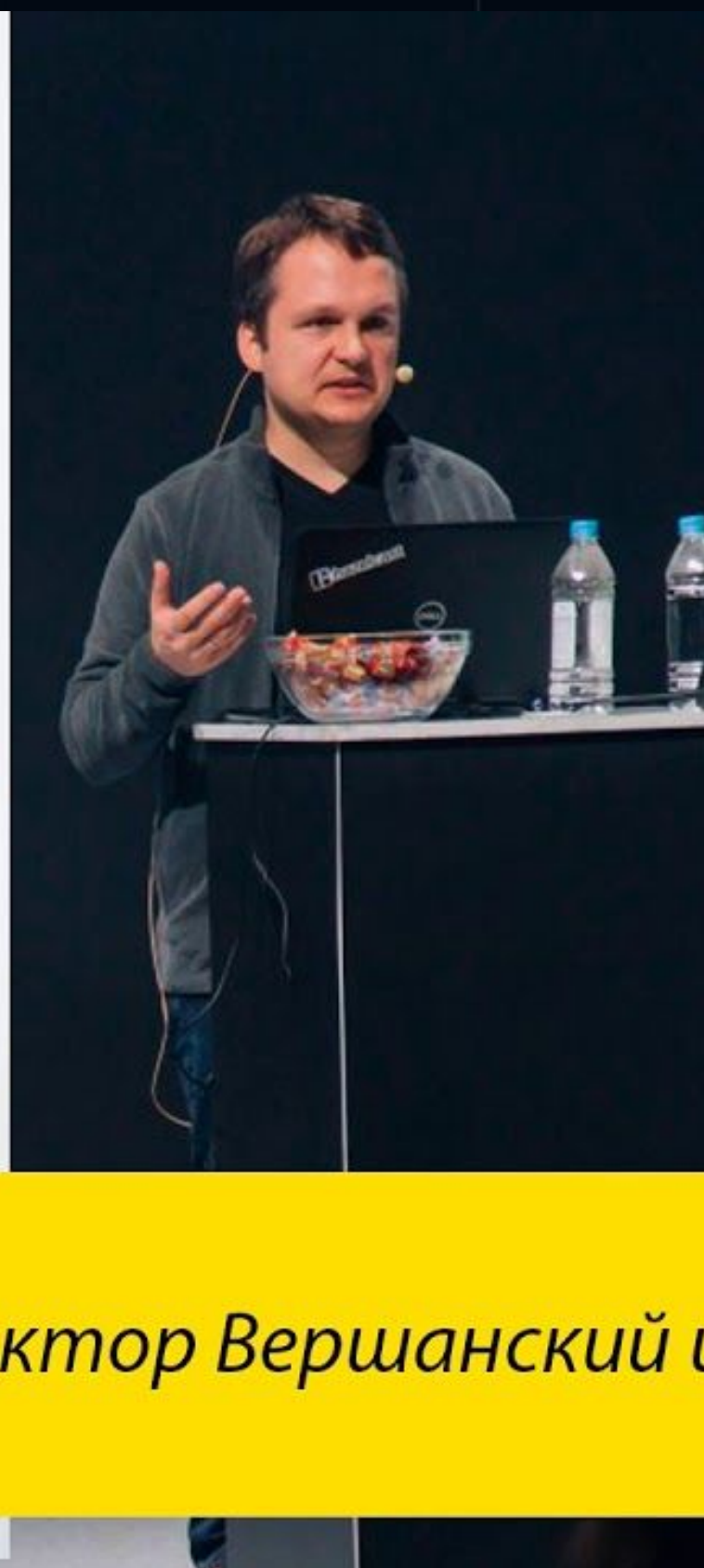


**Андрей Печуров**  
Hazelcast

История одной оптимизации  
производительности  
Node.js-библиотеки







*#11 - Виктор Вершанский и Андрей Печкуров [Ламповый]*







Andrei Pechkurov

Reviewed 14 pull requests in 3 repositories

[hazelcast/hazelcast](#)

[nodejs/node](#)

async\_hooks: add AsyncLocal API mk2

async\_hooks: introduce async-storage API

async\_hooks: add executionAsyncResource

[hazelcast/hazelcast-client-protocol](#)

Created an issue in [nodejs/diagnostics](#) that received 30 comments

### ✓ [async hooks] proposal for standard CLS API - request for feedback

Hi guys, I believe that upcoming `executionAsyncResource()` function ([nodejs/node#30959](#)) will allow building a simple and robust CLS API as a part of a...

30 comments



9 pull requests

2023

3 pull requests

Dec 25

2022

Dec 21

2021

Dec 16

2 pull requests

2020

2019

Dec 18

2018



github.com/nodejs/node/pull/26540

## async-hooks: introduce async-storage API #26540

vdeturckheim wants to merge 1 commit into `nodejs:master` from `vdeturckheim:async_storage`



**puzpuzpuz** commented on Mar 6, 2020

Any chance this will be backported to v12?

It would be great to have `AsyncLocalStorage` in v12, so I'm all for backporting it.

Note: `executionAsyncResource` ([#30959](#)) has to be backported first.

cc [@Qard](#) [@vdeturckheim](#)





# Class: AsyncLocalStorage

## ► History

This class creates stores that stay coherent through asynchronous operations.

While you can create your own implementation on top of the `node:async_hooks` module, `AsyncLocalStorage` performant and memory safe implementation that involves significant optimizations that are non-obvious to impl

The following example uses `AsyncLocalStorage` to build a simple logger that assigns IDs to incoming HTTP request messages logged within each request.

```
const http = require('node:http');  
const { AsyncLocalStorage } = require('node:async_hooks');  
  
const asyncLocalStorage = new AsyncLocalStorage();
```



# Девшахта №100: Асинхронный контекст



Девшахта №100: Асинхронный контекст



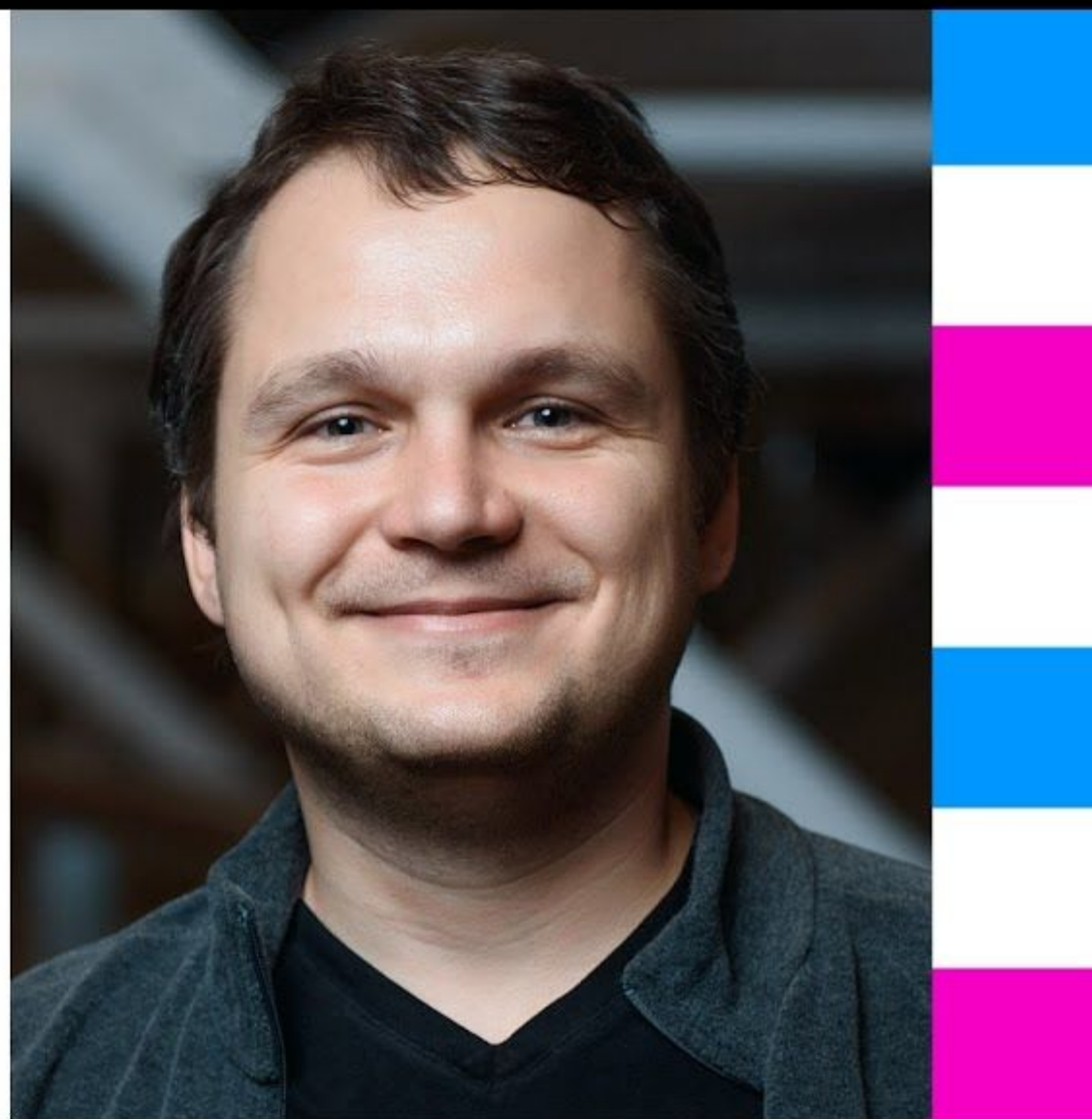


# Алгоритмы консенсуса. При чём тут Node.js?



**Андрей Печуров**  
Hazelcast

Алгоритмы консенсуса. При чём тут Node.js?





2019





**ТОЛЬКО МОЯ ЗАДАЧА ТАК И  
ОСТАЛАСЬ БЕЗ РЕШЕНИЯ**





**прошлое**





**прошлое**



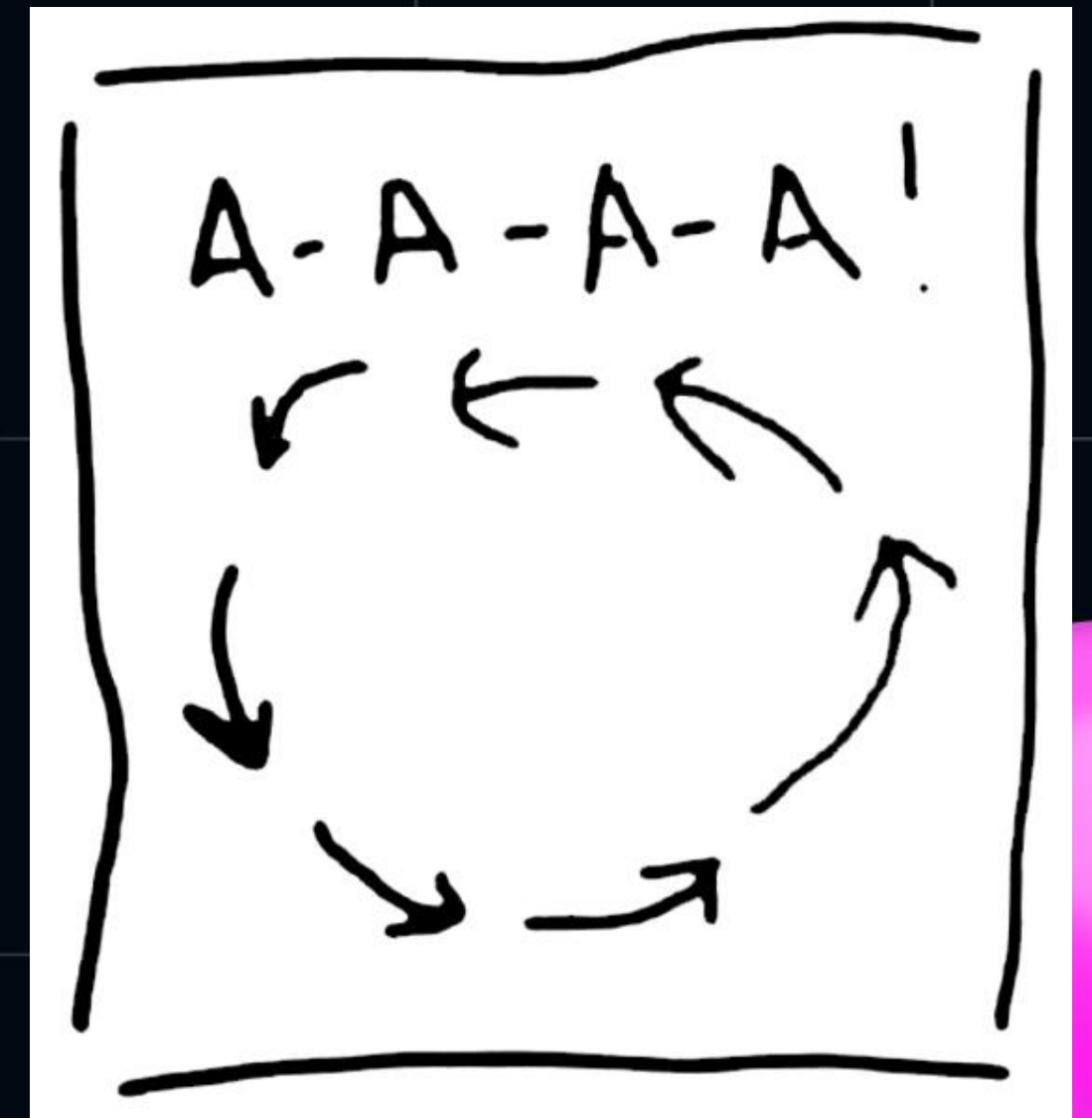


- КОНТЕКСТ постановки задачи
- формулировка проблематики
- **про что уже рассказывал по теме**





- контекст постановки задачи
- формулировка проблематики
- **про что уже рассказывал по теме**







**BrendanEich** ✓

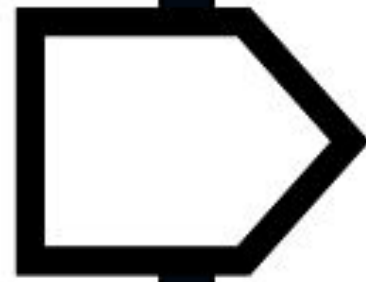
@BrendanEich

Replying to [@BrendanEich](#) [@rauschma](#) and [@IndieScripter](#)

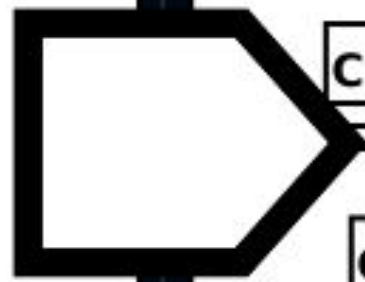
If I didn't have "Make it look like Java" as an order from management, \*and\* I had more time (hard to unconfound these two causal factors), then I would have preferred a Self-like "everything's an object" approach: no Boolean, Number, String wrappers. No undefined and null. Sigh.



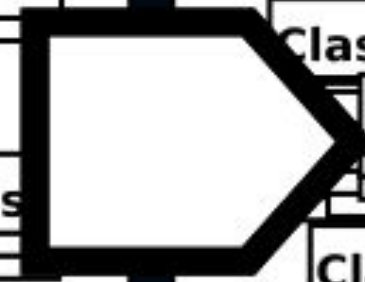
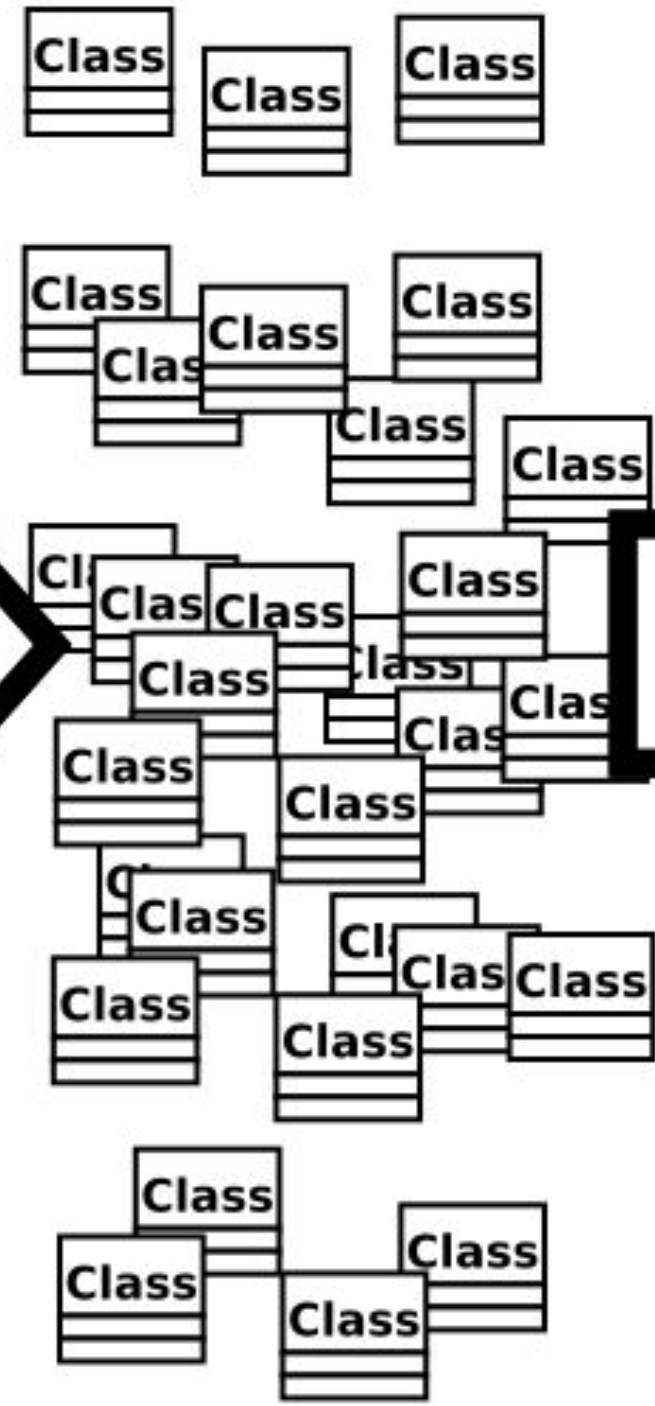
1 day



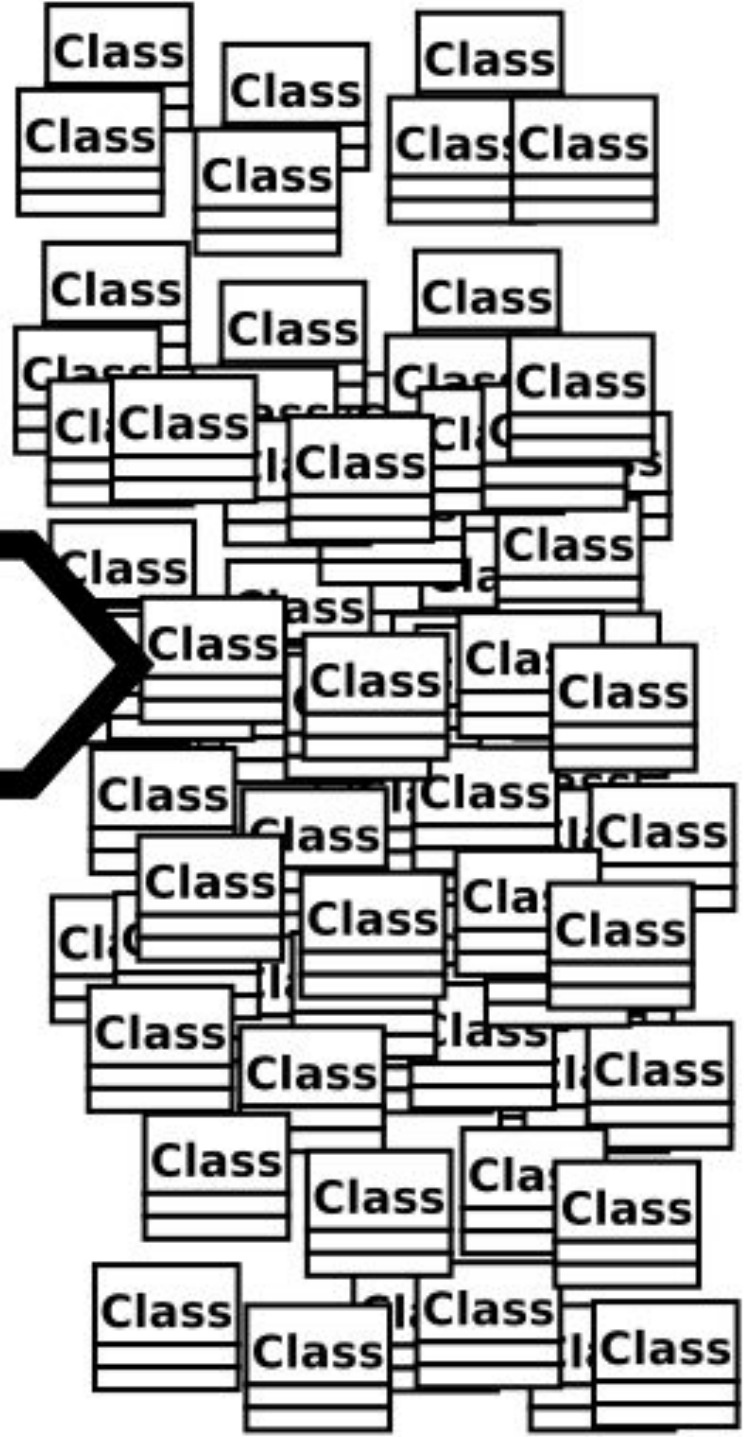
2 days



3 days

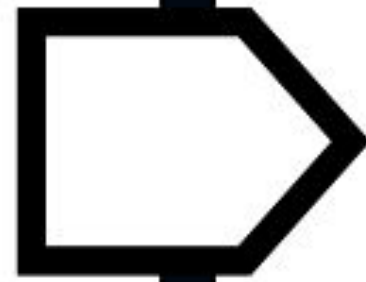


3 month

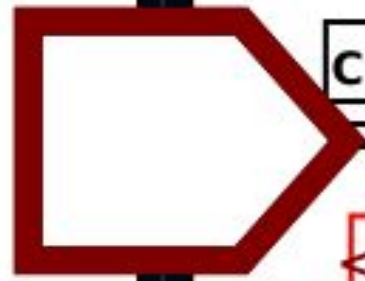
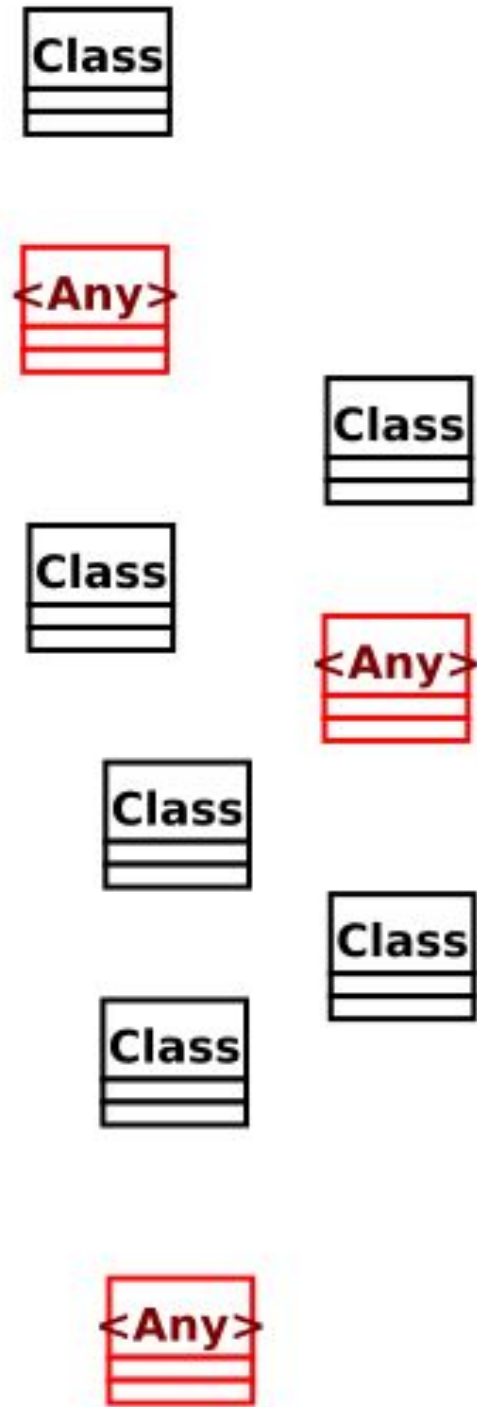




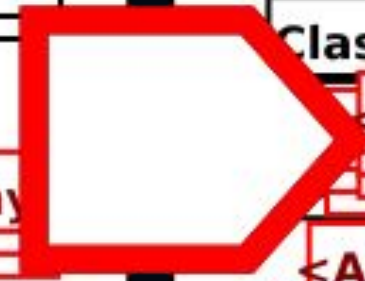
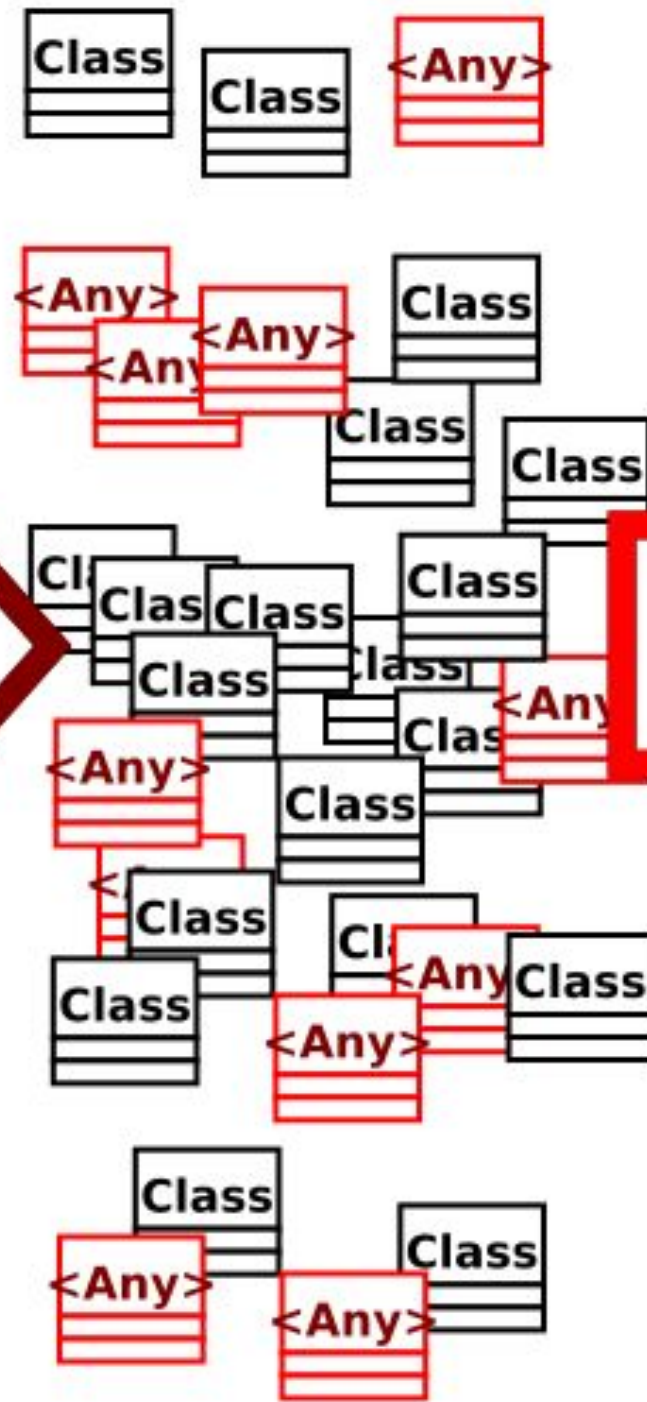
1 day



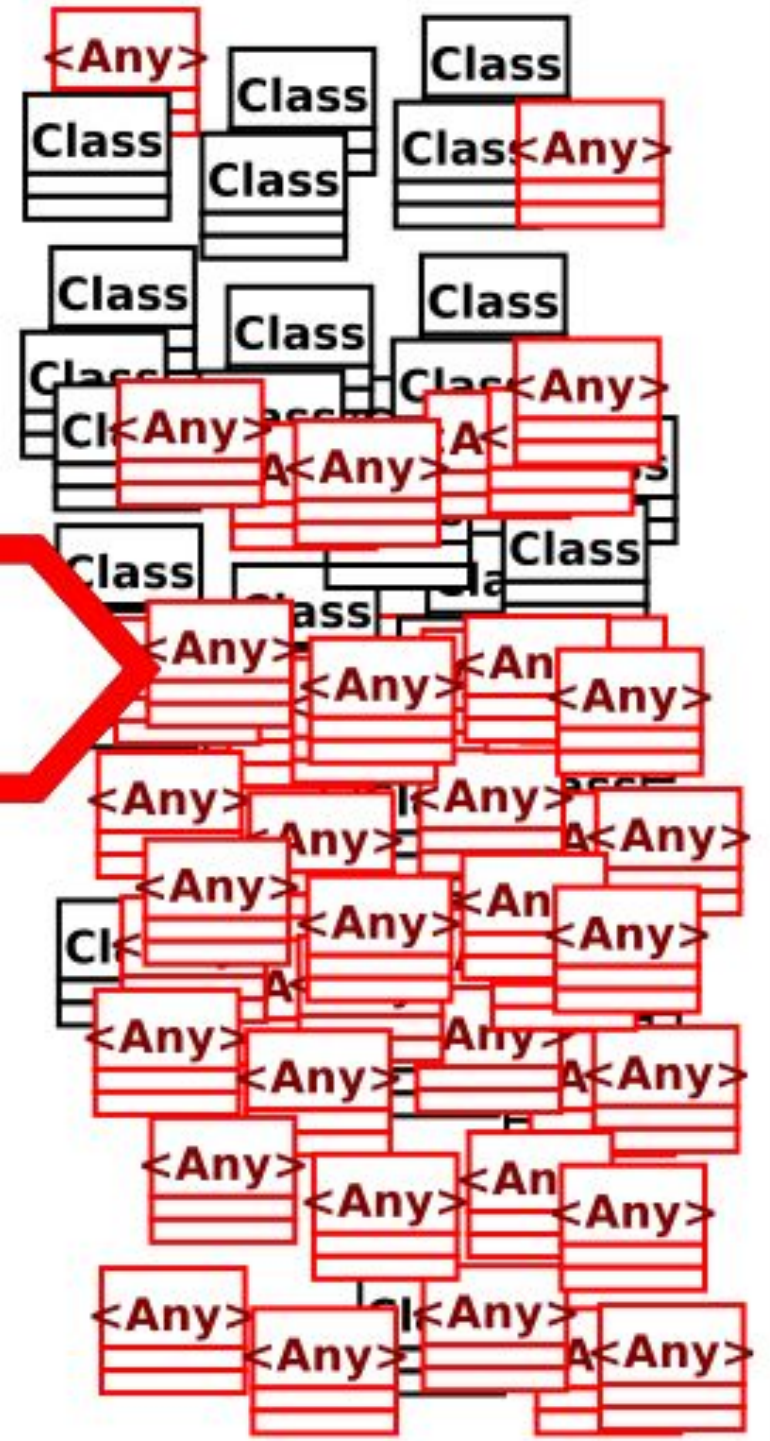
2 years



3 years



N years







**BrendanEich** ✓

@BrendanEich

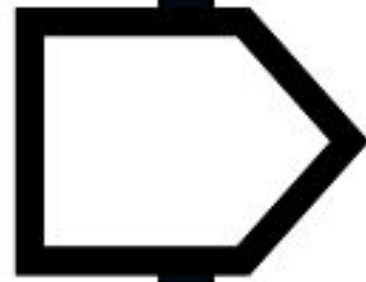
**и всё это объекты**

Replying to [@BrendanEich](#) [@rauschma](#) and [@IndieScripter](#)

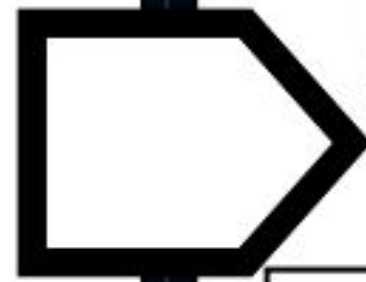
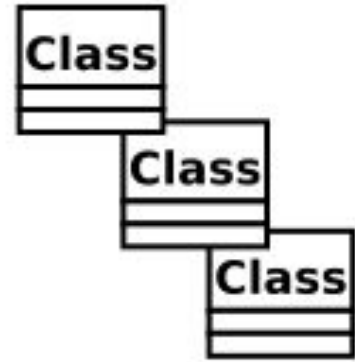
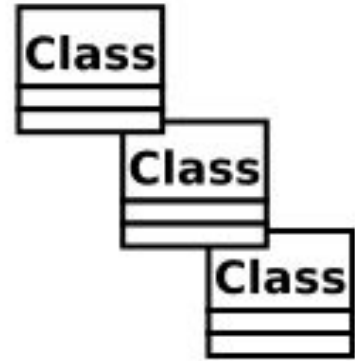
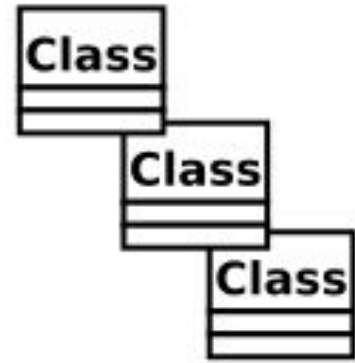
If I didn't have "Make it look like Java" as an order from management, \*and\* I had more time (hard to unconfound these two causal factors), then I would have preferred a Self-like "everything's an object" approach: no Boolean, Number, String wrappers. No undefined and null. Sigh.



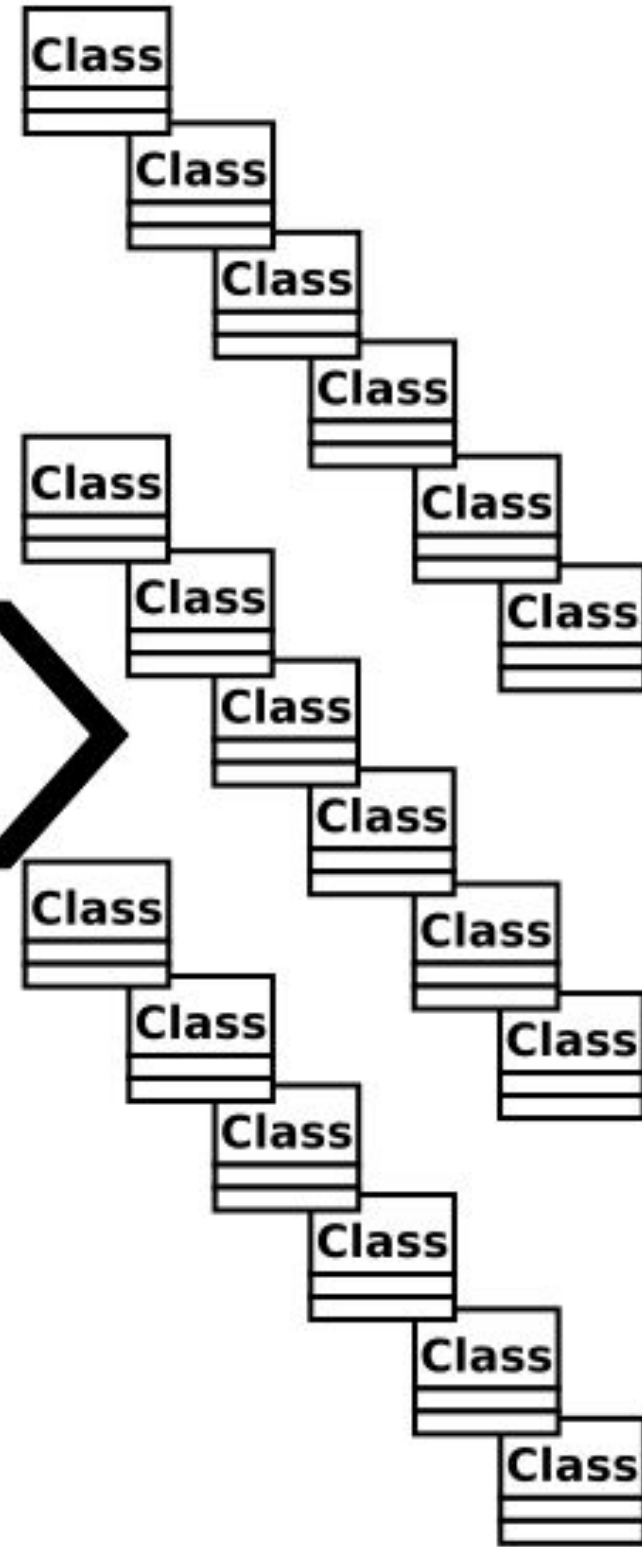
1 year



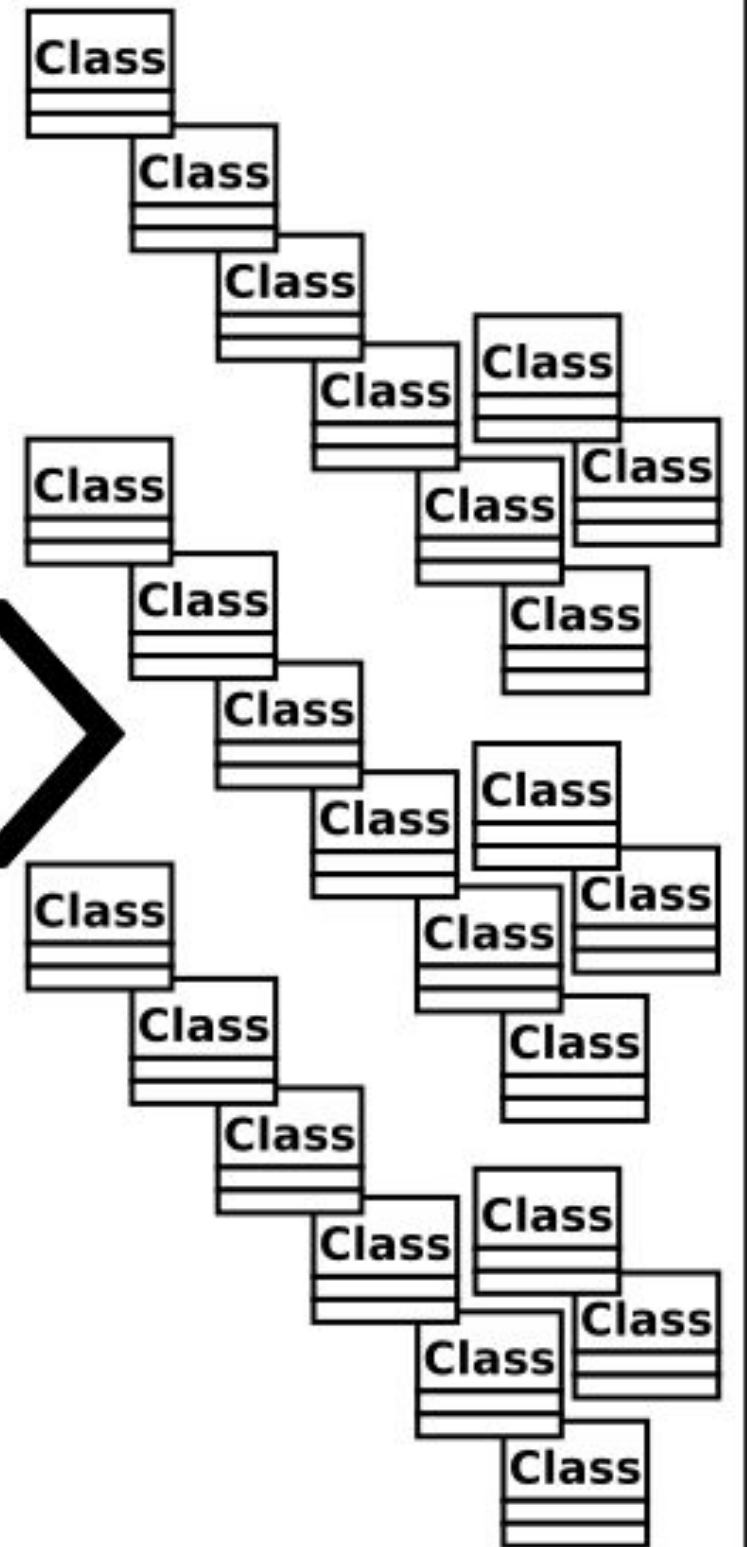
2 years



3 years



N years





> next

◀ ◻ *MyConstructor* {state: 3} ⓘ

state: 3

◻ *\_\_proto\_\_*:

state: 2

◻ *\_\_proto\_\_*:

state: 1

▶ *\_\_proto\_\_*: Object



**так как у меня уже был  
опыт с декораторами**





# context-dive

## Dive to async code with Context (v2)

Using this module you are able to achieve the following functionality:

1. CLS/TLS : Continuation/Thread Local Storage ([wikipedia link](#)).
2. Performance measuring, based on the top of **Dived Context** we are running in.
3. Meaningfull UncaughtException and UnhandledRejection extras, based on CLS.

... also there is plan to develop bit more rich functionality ...

So this module allows you to wrap some execution context context via [async\\_hooks](#). The main Idea is the wrapped **execution context** must be a function, cause otherwise we will be unable to handle everything through callbacks. Core concept is about that if we have some attributes of wrapped function as callbacks, so we will wrap them too, and therefore we will be able to track the context back through that callbacks too.



# context-dive

## про execution flow

### Dive to async code with Context (v2)

Using this module you are able to achieve the following functionality:

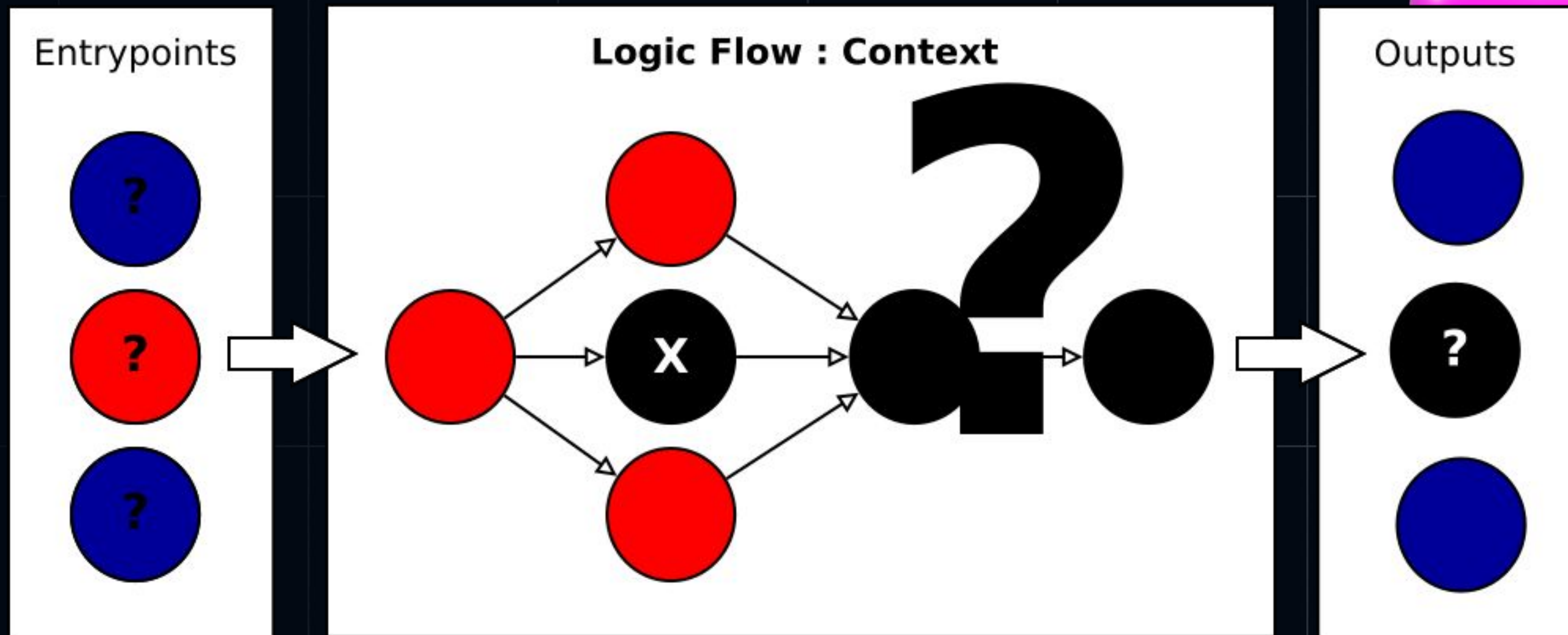
1. CLS/TLS : Continuation/Thread Local Storage ([wikipedia link](#)).
2. Performance measuring, based on the top of **Dived Context** we are running in.
3. Meaningfull UncaughtException and UnhandledRejection extras, based on CLS.

... also there is plan to develop bit more rich functionality ...

So this module allows you to wrap some execution context context via [async\\_hooks](#). The main Idea is the wrapped **execution context** must be a function, cause otherwise we will be unable to handle everything through callbacks. Core concept is about that if we have some attributes of wrapped function as callbacks, so we will wrap them too, and therefore we will be able to track the context back through that callbacks too.



# В различных по потокам данных





**решил упорядочить  
не потоки данных**





**решил упорядочить  
не потоки данных  
а сами данные**



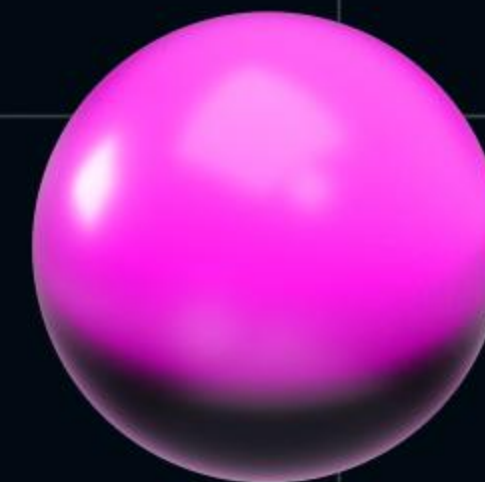


**решил упорядочить  
не потоки данных  
а сами данные**





**при помощи**  
**Prototype Chain**





- 3
- 1
- 2
- ...

# Inheritance in JavaScript : Factory of Constructors with Prototype Chain : point of view from boring nerd

#javascript #inheritance #prototype

went Feb 25 Originally published at [Medium](#) • 18 min read Edit Manage



 **went**  
JavaScript enthusiast. Computerman.  
Teapot Systems Administrator.  
[Edit profile](#)  
JOINED  
Jan 8, 2020

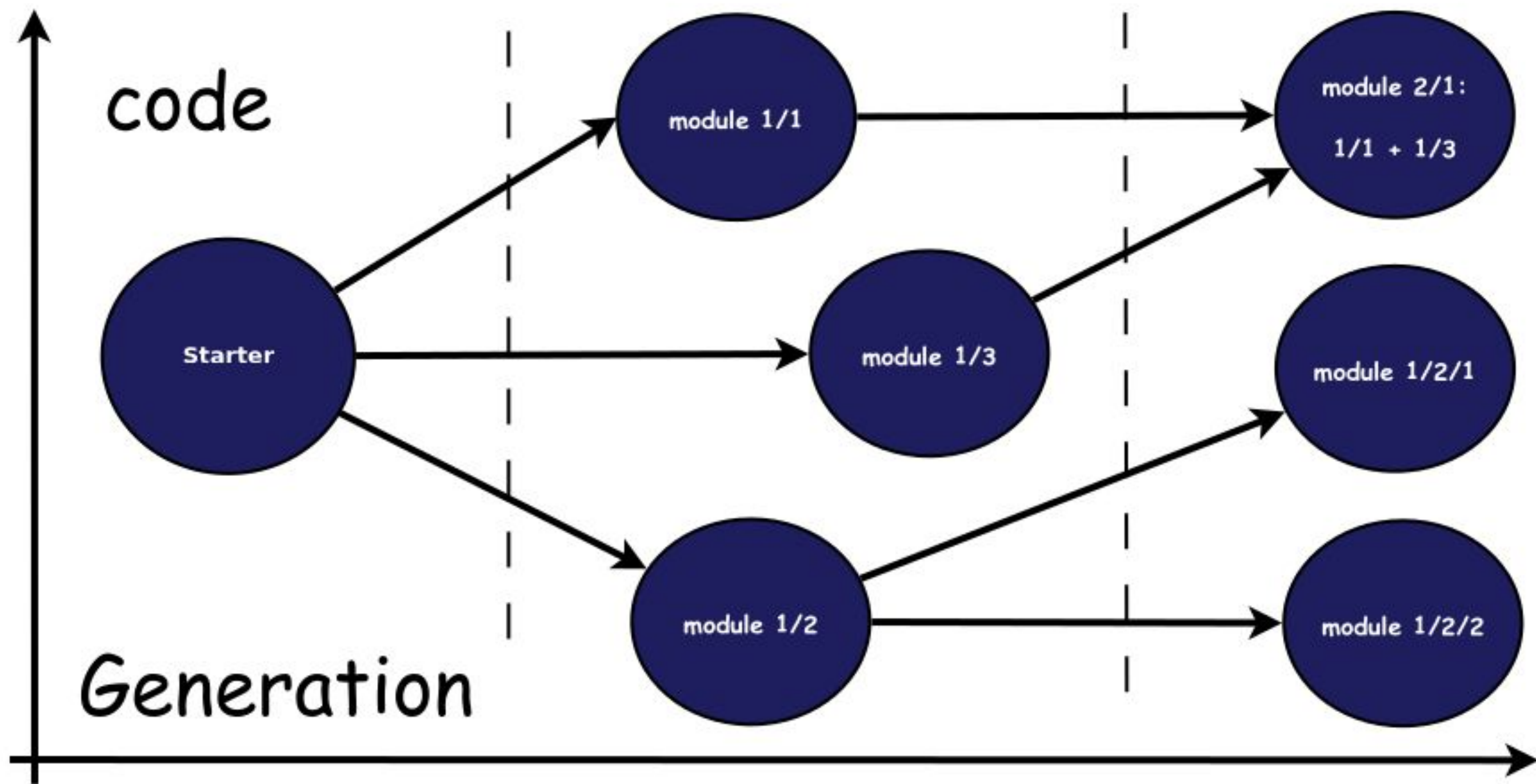
- ### More from went
- Dead Simple type checker for JavaScript  
#javascript #typechek #inritance
  - Architecture of Prototype Inheritance in JavaScript  
#inheritance #architecture  
#javascript



при помощи  
**Prototype Chain**









# mnemonica TS

0.9.955 • Public • Published 3 days ago

[Readme](#)

[Code](#) Beta

[0 Dependencies](#)

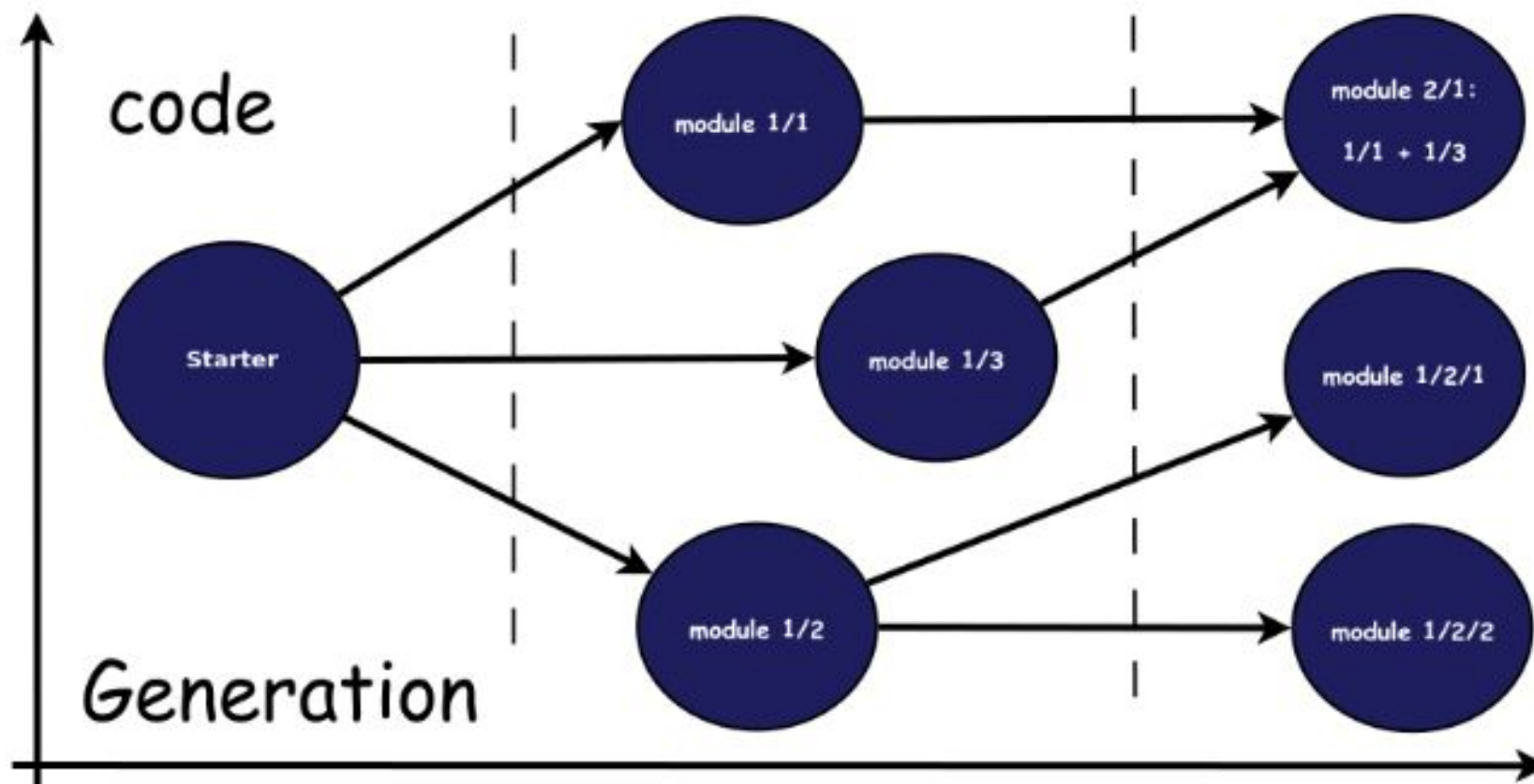
[0 Dependents](#)

[136 Versions](#)

## mnemonica is

abstract technique that aids information retention : instance inheritance system

... allows us to make inherited descriptions of mappings of transformations from predecessor structured data types to the successors, as if it was math  $f(x) \Rightarrow y$  ... and we will use `this` keyword as a persistent data structure where we will apply that transformations



### Install

```
> npm i mnemonica
```

### Repository

[github.com/wentout/mnemonica](https://github.com/wentout/mnemonica)

### Homepage

[github.com/wentout/mnemonica#read...](https://github.com/wentout/mnemonica#readme)

### Weekly Downloads

26



### Version

0.9.955

### License

MIT

### Unpacked Size

120 kB

### Total Files

80

### Issues

0

### Pull Requests

0





Chronotope: await  
Eloquent.Errors

#### Приглашенные эксперты



Дмитрий Махнёв  
JetBrains

#### Спикеры



Виктор Вершанский



2020





# Оптимизация синхронной асинхронности



2021 PITER

**Дмитрий Махнёв**

JetBrains

Оптимизация синхронной  
асинхронности





# typeomatica TS

0.3.31 • Public • Published 21 days ago

[Readme](#)

[Code](#) Beta

[0 Dependencies](#)

[0 Dependents](#)

[23 Versions](#)

## Type ø matica

coverage 100%

license MIT version v0.3.31 last commit october

**\$ npm install typeomatica**

This package is a part of **mnemonica** project.

Strict Types checker for objects which represent Data Types.

## how it works

see `test/index.ts`

```
class SimpleBase extends BasePrototype {  
    stringProp = '123';  
};
```

### Install

```
> npm i typeomatica
```

### Repository

[github.com/wentout/typeomatica](https://github.com/wentout/typeomatica)

### Homepage

[github.com/wentout/typeomatica#read...](https://github.com/wentout/typeomatica#readme)

### Weekly Downloads

0

Version

0.3.31

License

MIT

Unpacked Size

49.2 kB

Total Files

42



# Строгая типизация в JavaScript



2021 PITER

**Виктор Вершанский**

DataArt

Strict Types in JavaScript





2021

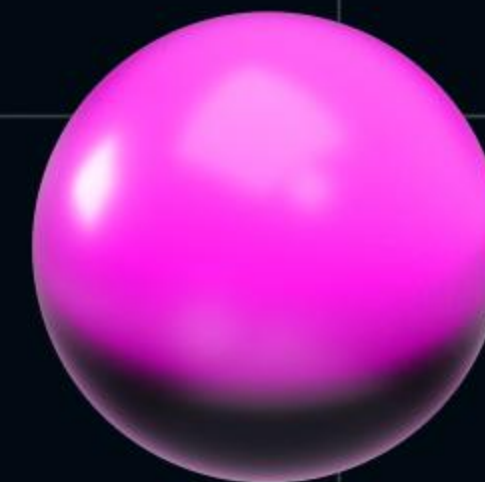








**КОД**





**КОД**





- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- **как создаётся код для решения**





- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- **как создаётся код для решения**





```
declare function remapKeys
```

```
  <  
    T extends number,  
    RR extends string,  
    O extends Record<string, T>,  
    R extends Record<string, RR>  
  >
```

```
(obj: O, remap: R): {
```

```
  [K in keyof O as K extends keyof R ? R[K] : K]: O[K]  
}
```

```
var res = remapKeys({ obj: 1, a: 2 }, { obj: 'newObj' })
```



```
declare function remapKeys
```

```
· <  
· · T extends number,  
· · RR extends string,  
· · O extends Record<string, T>,  
· · R extends Record<string, RR>  
· >
```

```
(obj: O, remap: R): {
```

```
· [K in keyof O as K extends keyof R ? R[K] : K]: O[K]  
}
```

```
var res = remapKeys({ obj: 1, a: 2 }, { obj: 'newObj' })
```





- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- **как создаётся код для решения**







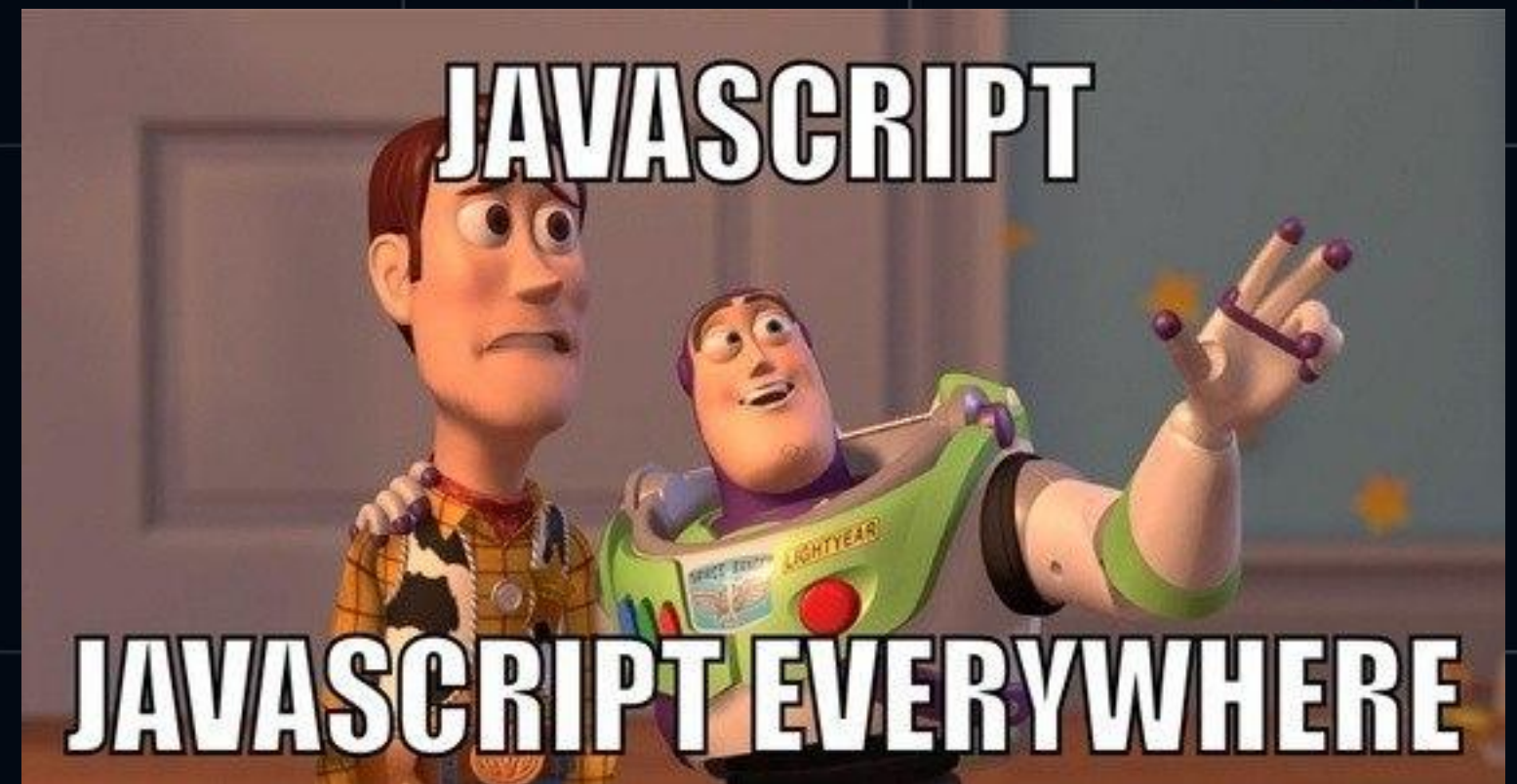


**для начала нужно типизировать  
функциональный конструктор**





- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- **как создаётся код для решения**






```
1 type Proto<P, T> = Pick<P, Exclude<keyof P, keyof T>> & T;
2
3 const CstrFn = function <R>(param: R) {
4   Object.assign(this, param);
5 } as {
6   (): void
7   new <
8     T extends object,
9     S extends Proto<T, typeof CstrFn.prototype>>(param: T): {
10     [key in keyof S]: S[key]
11   }
12   prototype: {
13     m: number
14   }
15 };
16 CstrFn.prototype.m = 2
17
18 const s = new CstrFn({ m: '1', s: 1 });
19 console.log(s);
```



```
1 type Proto<P, T> = Pick<P, Exclude<keyof P, keyof T>> & T;
2
3 const CstrFn = function <R>(param: R) {
4   Object.assign(this, param);
5 } as {
6   (): void
7   new <
8     T extends object,
9     S extends Proto<T, typeof CstrFn.prototype>>(param: T): {
10     [key in keyof S]: S[key]
11   }
12   prototype: {
13     m: number
14     const s: {
15       s: number;
16       m: number;
17     }
18 };
19 CstrFn = new CstrFn({ m: '1', s: 1 });
20 console.log(s);
```





**то есть да,  
такая вот простыня**






```
1 type Proto<P, T> = Pick<P, Exclude<keyof P, keyof T>> & T;
2
3 const CstrFn = function <R>(param: R) {
4   Object.assign(this, param);
5 } as {
6   (): void
7   new <
8     T extends object,
9     S extends Proto<T, typeof CstrFn.prototype>>(param: T): {
10     [key in keyof S]: S[key]
11   }
12   prototype: {
13     m: number
14   }
15 };
16 CstrFn.prototype.m = 2
17
18 const s = new CstrFn({ m: '1', s: 1 });
19 console.log(s);
```

**НО ОНА В САМОМ ДЕЛЕ НУЖНА**





```
1 type Proto<P, T> = Pick<P, Exclude<keyof P, keyof T>> & T;
2
3 const CstrFn = function <R>(param: R) {
4   Object.assign(this, param);
5 } as {
6   (): void
7   new <
8     T extends object,
9     S extends Proto<T, typeof CstrFn.prototype>>(param: T): {
10     [key in keyof S]: S[key]
11   }
12   prototype: {
13     m: number
14     const s: {
15       s: number;
16       m: number;
17     }
18 };
19 CstrFn = new CstrFn({ m: '1', s: 1 });
20 console.log(s);
```



**и ещё инструментарий :**





# mnemonica TS

0.9.955 • Public • Published 3 days ago

[Readme](#)

[Code](#) Beta

[0 Dependencies](#)

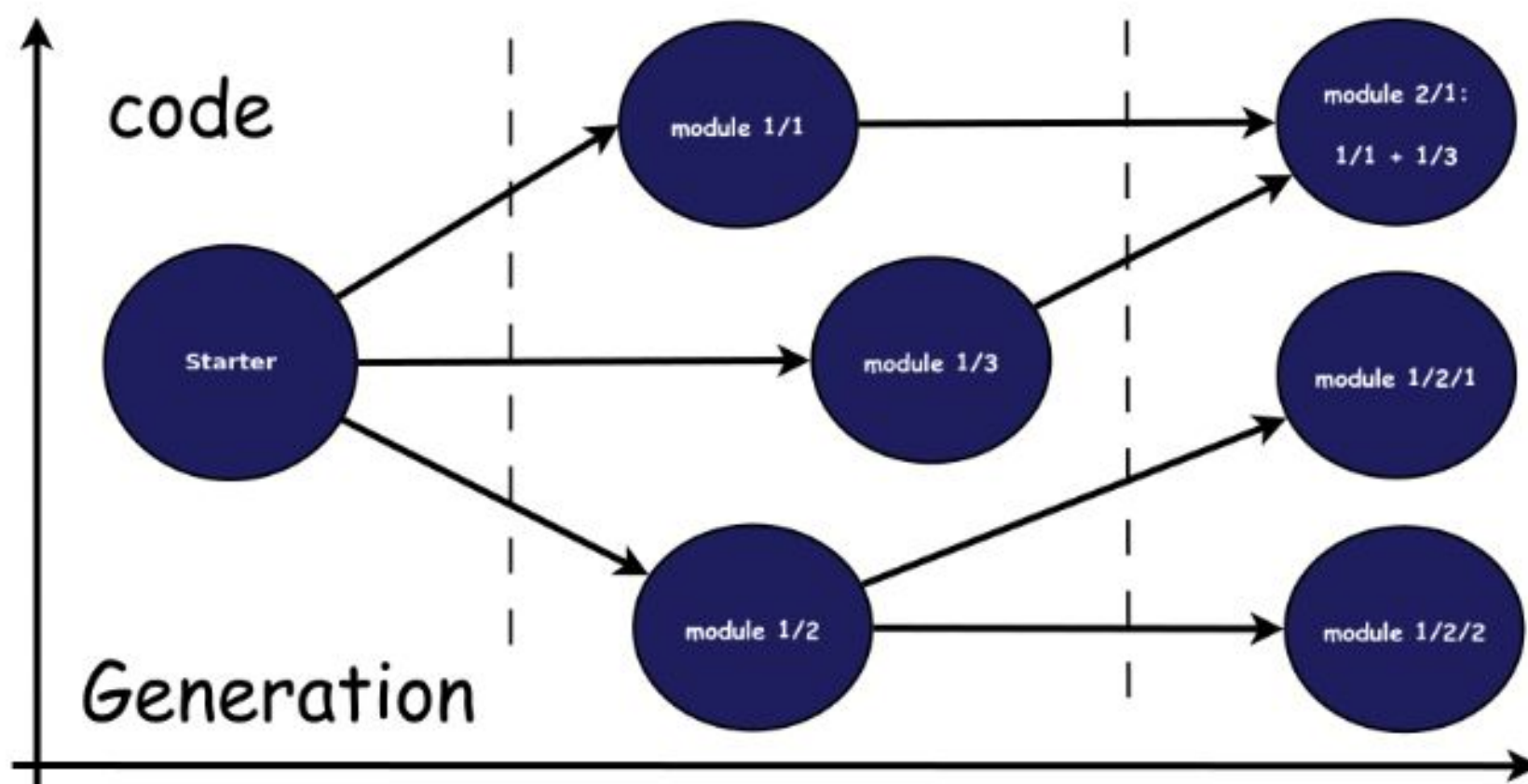
[0 Dependents](#)

[136 Versions](#)

## mnemonica is

abstract technique that aids information retention : instance inheritance system

... allows us to make inherited descriptions of mappings of transformations from predecessor structured data types to the successors, as if it was math  $f(x) \Rightarrow y$  ... and we will use `this` keyword as a persistent data structure where we will apply that transformations



### Install

```
> npm i mnemonica
```

### Repository

[github.com/wentout/mnemonica](https://github.com/wentout/mnemonica)

### Homepage

[github.com/wentout/mnemonica#read...](https://github.com/wentout/mnemonica#readme)

### Weekly Downloads

26



### Version

0.9.955

### License

MIT

### Unpacked Size

120 kB

### Total Files

80

### Issues

0

### Pull Requests

0

TS test-example.ts X

core > test-ts > TS test-example.ts > ...

```
1  import { define } from '..';
2
3  const FirstType = define('SomeType', function(this: {
4    |   first: 'FirstType',
5  }) {
6    |   this.first = 'FirstType';
7  });
8
9  const SecondType = FirstType.define('SecondType', function(this: {
10 |   first: undefined,
11 |   second: string,
12 | }) {
13 |   this.first = undefined;
14 |   this.second = 'SecondType';
15 | });
```



TS test-example.ts ✕

core > test-ts > TS test-example.ts > ...

```
16
17  const first = new FirstType();
18
19  type TSecondInstance =
20  |> |> InstanceType<typeof SecondType>;
21
22  const second = new first.SecondType() as TSecondInstance;
23
24  // { first: undefined, second: "SecondType" }
25  console.log(second);
26
27
```





**ЧТО СЕЙЧАС**



**ЧТО СЕЙЧАС**





- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- как создаётся код для решения
- **как развивается решение: что сделано**



2023





2023



# Типы в прототипах



Виктор  
Вершанский

**о чём там речь**





# О ЧЁМ ТАМ РЕЧЬ



Issue Type\* 🔴 Bug ⓘ

Some issue types are unavailable due to incompatible field configuration and/or workflow associations.

---

Main **Additional**

Summary\*

Priority 🟢 Нормально ⓘ

Component/s

Assignee

Story Points

- Нормально
- 🚫 Очень срочно
- 🔴 Очень важно
- 🟡 Важно
- ⚪ Когда-нибудь
- ⚪ Маловажное возможное улуч...

press down to select.



TS mixWithProto.ts X



talks > 2023-05-HolyJS > examples > TS mixWithProto.ts > ...

```
1  type init = {  
2    s: number  
3    z: number  
4  }  
5  
6  type next = {  
7    s?: string  
8    m: boolean  
9  }  
10
```

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```



TS mixWithProto.ts ×



talks > 2023-05-HolyJS > examples > TS mixWithProto.ts > ...

```
22 type unit = proto & next
```

```
23
```

```
24 const aggregation: unit = {
```

```
25   → z: 123,
```

```
26   → s: 'x',
```

```
27   → m: true,
```

```
28 };
```

```
29   type sss = string | undefined
```

```
30 type sss = typeof aggregation.s
```

```
31
```

TS mixWithProto.ts X



talks > 2023-05-HolyJS > examples > TS mixWithProto.ts > ...

```
1  type init = {  
2    s: number  
3    z: number  
4  }  
5  
6  type next = {  
7    s?: string  
8    m: boolean  
9  }  
10
```

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```





TS mixWithProto.ts X



talks > 2023-05-HolyJS > examples > TS mixWithProto.ts > ...

10

11

12

13

```
type proto = {  
  z: number;  
}
```

14

```
type proto = Pick<
```

15

```
  init,
```

16

```
  Exclude<
```

17

```
    keyof init,
```

18

```
    keyof next
```

19

```
>>
```

TS function\_construct\_typed.ts ×

examples > TS function\_construct\_typed.ts > ...

```
1 function OtherConstruct(this: { field: number }) {}
2 OtherConstruct.prototype = {
3   otherField: true
4 }
5
6 type Proto<P, T> = Pick<P, Exclude<keyof P, keyof T>> & T;
7
8 const define = function <P extends object, T>(Cstr: { (this: T): void }, proto: P) {
9   const MyConstructor = function (): Proto<P, T> {
10     return new Cstr;
11   };
12   Object.setPrototypeOf(MyConstructor.prototype, proto);
13   return MyConstructor;
14 };
15
16 const myConstruct = define(OtherConstruct, { otherField: true });
17
18 const myConstructedItem = myConstruct();
19
20 console.log(myConstructedItem);
21
```



examples &gt; TS function\_construct\_typed.ts &gt; ...

```
1 function OtherConstruct(this: { field: number }) {}
2 OtherConstruct.prototype = {
3   otherField: true
4 }
5
6 type Proto<P, T> = Pick<P, Exclude<keyof P, keyof T>> & T;
7
8 const define = function <P extends object, T>(Cstr: { (this: T): void }, proto: P) {
9   const MyConstructor = function (): Proto<P, T> {
10     return new Cstr;
11   };
12   Object.setPrototypeOf(MyConstructor.prototype, proto);
13   return { myConstructedItem: Proto<{
14     otherField: boolean;
15   }, {
16     field: number;
17   }>
18     t, { otherField: true });
19
20 console.log(myConstructedItem);
21
```



examples &gt; TS function\_construct\_typed.ts &gt; ...

```
1 function OtherConstruct(this: { field: number }) {}
2 OtherConstruct.prototype = {
3   otherField: true
4 }
5
6 type Proto<P, T> = Pick<P, Exclude<keyof P, keyof T>> & T;
7
8 const define = function <P extends object, T>(Cstr: { (this: T): void }, proto: P) {
9   const MyConstructor = function (): Proto<P, T> {
10     return new Cstr;
11   };
12   Object.setPrototypeOf(MyConstructor.prototype, proto);
13   return {
14     myConstructedItem: Proto<{
15       otherField: boolean;
16       field: number;
17     }>(Cstr, { otherField: true });
18 };
19
20 const myConstructedItem = myConstruct();
21 console.log(myConstructedItem);
```



examples &gt; TS function\_construct\_typed.ts &gt; ...

```
1 function OtherConstruct(this: { field: number }) {}
2 OtherConstruct.prototype = {
3   otherField: true
4 }
5
6 type Proto<P, T> = Pick<P, Exclude<keyof P, keyof T>> & T;
7
8
9
10
11
12 Object.setPrototypeOf(MyConstructor.prototype, proto);
13 re const myConstructedItem: Proto<{
14   otherField: boolean;
15   }, {
16     field: number;
17   }>
18 const myConstructedItem = myConstruct();
19
20 console.log(myConstructedItem);
21
```

**но есть проблемка ...**





# mnemonica TS

0.9.955 • Public • Published 3 days ago

[Readme](#)

[Code](#) Beta

[0 Dependencies](#)

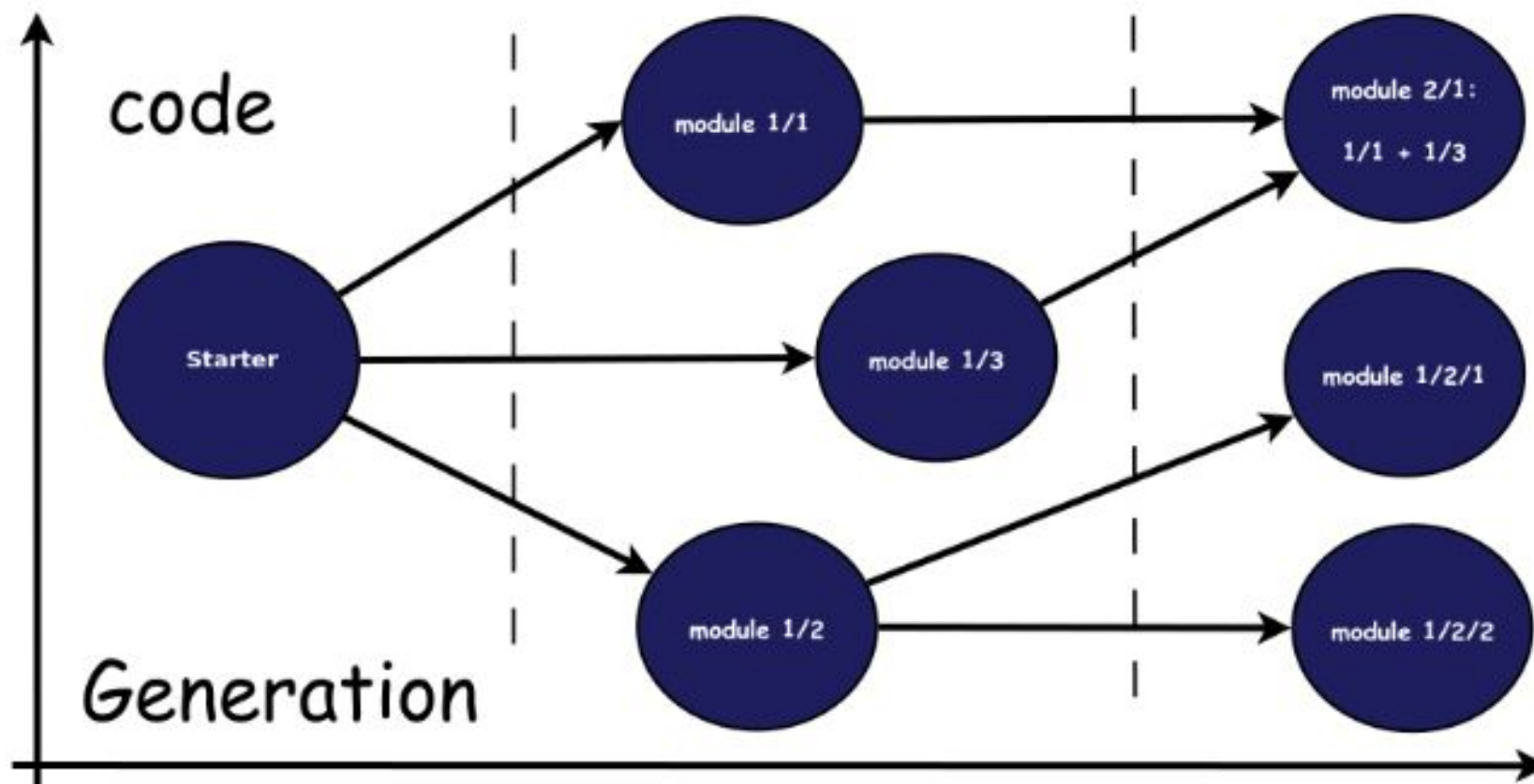
[0 Dependents](#)

[136 Versions](#)

## mnemonica is

abstract technique that aids information retention : instance inheritance system

... allows us to make inherited descriptions of mappings of transformations from predecessor structured data types to the successors, as if it was math  $f(x) \Rightarrow y$  ... and we will use `this` keyword as a persistent data structure where we will apply that transformations



### Install

```
> npm i mnemonica
```

### Repository

[github.com/wentout/mnemonica](https://github.com/wentout/mnemonica)

### Homepage

[github.com/wentout/mnemonica#read...](https://github.com/wentout/mnemonica#readme)

### Weekly Downloads

26



### Version

0.9.955

### License

MIT

### Unpacked Size

120 kB

### Total Files

80

### Issues

0

### Pull Requests

0

TS test-example.ts ✕

core > test-ts > TS test-example.ts > ...

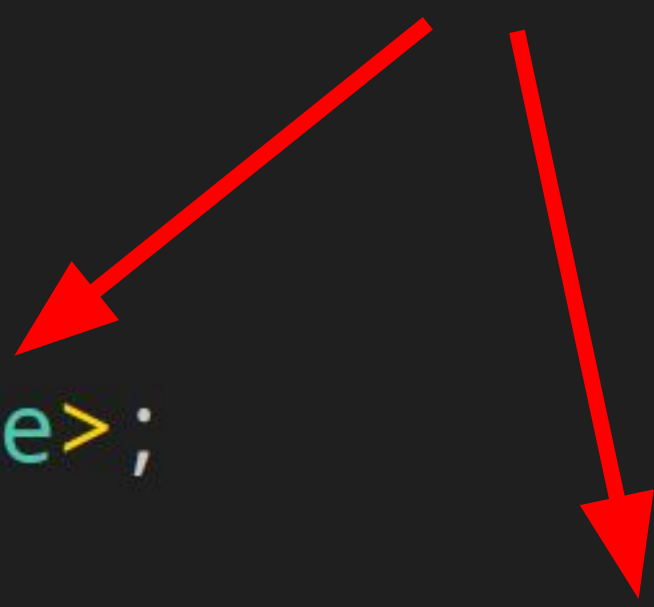
```
16
17  const first = new FirstType();
18
19  type TSecondInstance =
20  |> |> InstanceType<typeof SecondType>;
21
22  const second = new first.SecondType() as TSecondInstance;
23
24  // { first: undefined, second: "SecondType" }
25  console.log(second);
26
27
```



TS test-example.ts ✕

core > test-ts > TS test-example.ts > ...

```
16
17  const first = new FirstType();
18
19  type TSecondInstance =
20  |> |> InstanceType<typeof SecondType>;
21
22  const second = new first.SecondType() as TSecondInstance;
23
24  // { first: undefined, second: "SecondType" }
25  console.log(second);
26
27
```





TS test-example.ts ✕

core > test-ts > TS test-example.ts > ...

16

```
17 const first = new FirstType();
```

18

```
19 type TSecondInstance =
```

```
20     ↳     ↳ InstanceType<typeof SecondType>;
```

21

```
22 const second = new first.SecondType() as TSecondInstance;
```

23

```
24 // { first: undefined, second: "SecondType" }
```

```
25 console.log(second);
```

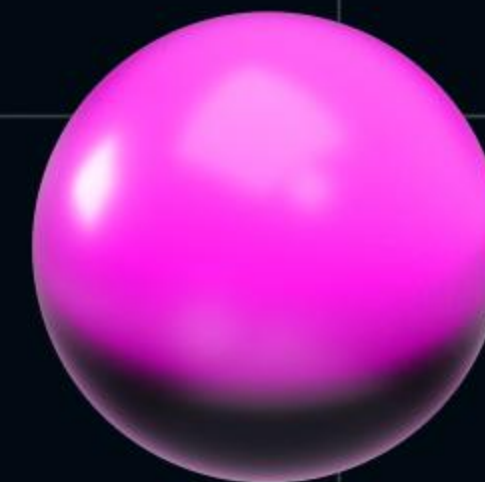
26

27





**... решение есть ...**



**... apply, call, bind ...**





TS test-example.ts ✕

core > test-ts > TS test-example.ts > ...

```
16
17  const first = new FirstType();
18
19  type TSecondInstance =
20  |> |> InstanceType<typeof SecondType>;
21
22  const second = new first.SecondType() as TSecondInstance;
23
24  // { first: undefined, second: "SecondType" }
25  console.log(second);
26
27
```



TS init.ts

✕

TS direct.ts

TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS &gt; examples &gt; TS init.ts &gt; ...

```
1  import { define } from 'mnemonica';
2
3  export const FirstType = define('SomeType', function(this: {
4  |   first: 'FirstType',
5  | }) {
6  |   this.first = 'FirstType';
7  | });
8
9  export const SecondType = FirstType.define('SecondType', function(this: {
10 |   first: undefined,
11 |   second: string,
12 | }) {
13 |   this.first = undefined;
14 |   this.second = 'SecondType';
15 | });
16
```



TS init.ts

TS direct.ts X

TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS direct.ts > ...

```
1 import { FirstType, SecondType } from './init';
2
3 const first = new FirstType();
4
5 type TSecondInstance = InstanceType<typeof SecondType>;
6 const second = new first.SecondType() as TSecondInstance;
7
8 // { first: undefined, second: string }
9 console.log(second);
10
11
12
```

TS init.ts

TS apply.ts ✕

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS apply.ts > ...

```
1  import { apply } from 'mnemonica';
2
3  import { FirstType, SecondType } from './init';
4
5  const first = new FirstType();
6
7  // { first: undefined, second: string }
8  const second = apply(first, SecondType);
9  console.log(second);
```

10

11



TS init.ts

TS apply.ts X

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS apply.ts > ...

```
1 import { apply } from 'mnemonica';
```

```
2
```

```
3 import const second: { it';
```

```
4   [x: string]: new () => unknown;
```

```
5   first: undefined;
```

```
6   second: string;
```

```
7 // { f }
```

```
8 const second = apply(first, SecondType);
```

```
9 console.log(second);
```

```
10
```

```
11
```

TS init.ts

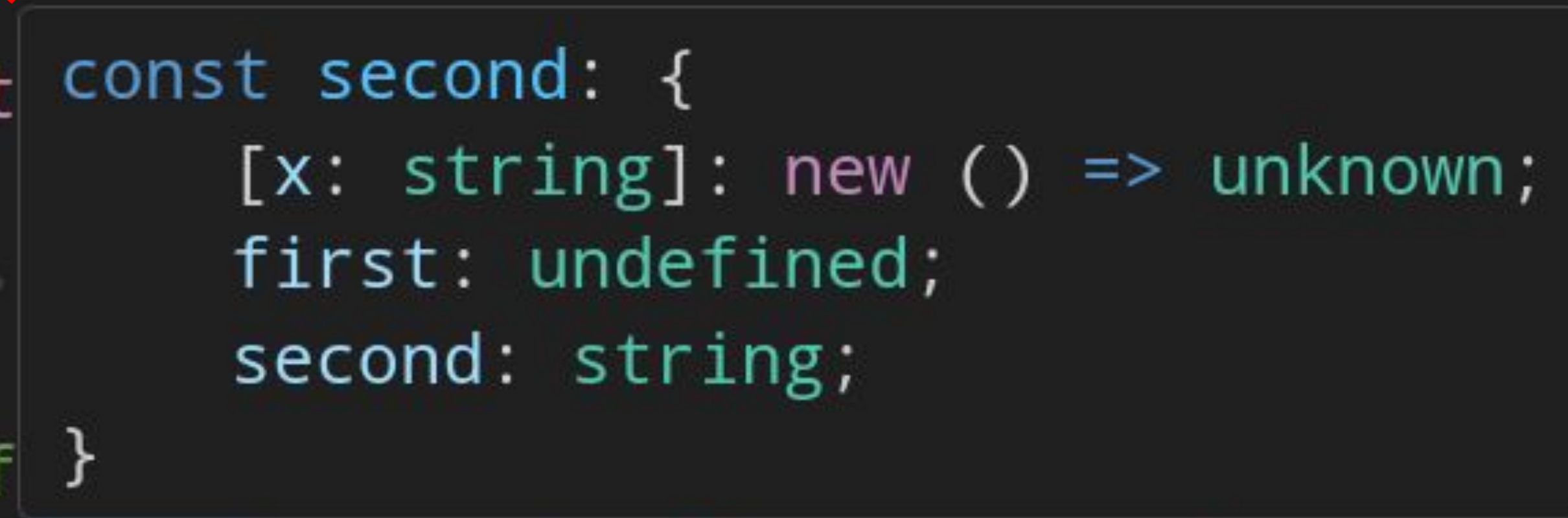
TS apply.ts X

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS apply.ts > ...

```
1 import { apply } from 'mnemonica';
2
3 import const second: {
4     [x: string]: new () => unknown;
5     first: undefined;
6     second: string;
7 } // { f
8 const second = apply(first, SecondType);
9 console.log(second);
10
11
```





TS init.ts

TS apply.ts

TS call.ts

×

TS bind.ts

2023-11-HolyJS > examples > TS call.ts > ...

```
1 import { call } from 'mnemonica';
```

```
2
```

```
3 import { FirstType, SecondType } from './init';
```

```
4
```

```
5 const first = new FirstType();
```

```
6
```

```
7 // { first: undefined, second: string }
```

```
8 const second = call(first, SecondType);
```

```
9 console.log(second);
```

```
10
```

```
11
```

TS init.ts

TS apply.ts

TS call.ts

×

TS bind.ts

code > talks > 2023-11-HolyJS > examples > TS call.ts > ...

```
1 import { call } from 'mnemonica';
```

```
2
```

```
3 import const second: { it';
```

```
4 [x: string]: new () => unknown;
```

```
5 const first: undefined;
```

```
6 second: string;
```

```
7 // { f }
```

```
8 const second = call(first, SecondType);
```

```
9 console.log(second);
```

```
10
```



TS init.ts

TS apply.ts

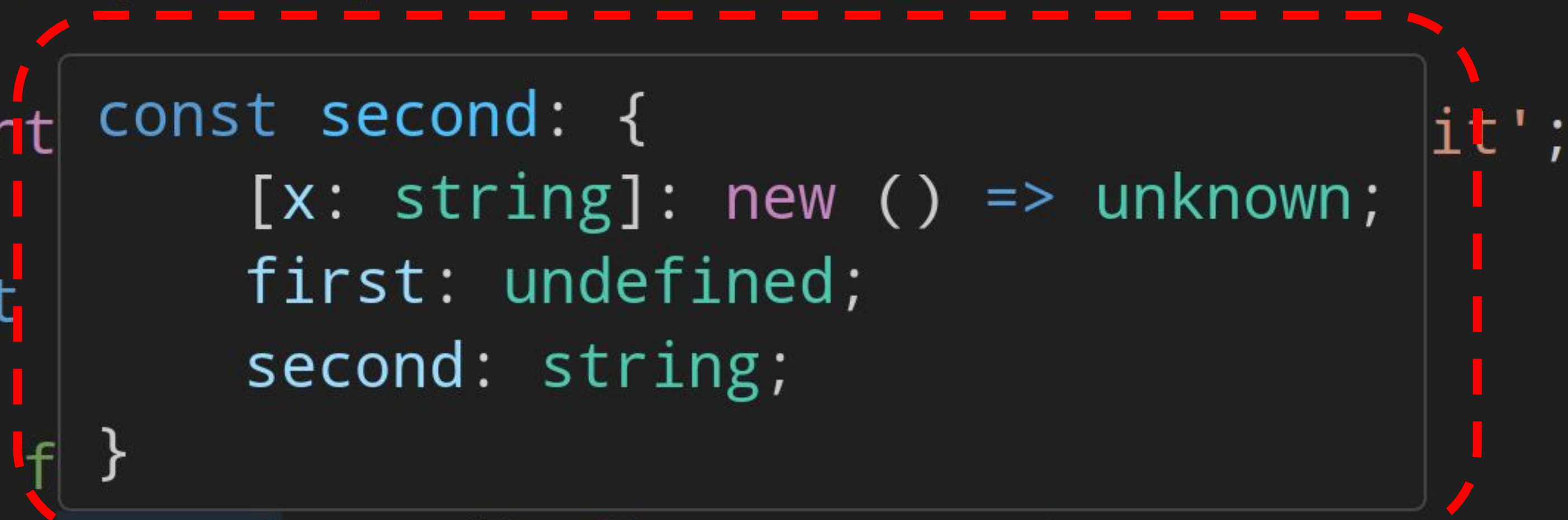
TS call.ts

×

TS bind.ts

code > talks > 2023-11-HolyJS > examples > TS call.ts > ...

```
1  import { call } from 'mnemonica';
2
3  import const second: {
4      [x: string]: new () => unknown;
5      first: undefined;
6      second: string;
7  }
8  // { f
9  const second = call(first, SecondType);
10 console.log(second);
```



TS init.ts

TS apply.ts

TS call.ts

TS bind.ts

×

2023-11-HolyJS > examples > TS bind.ts > ...

```
1  import { bind } from 'mnemonica';
2
3  import { FirstType, SecondType } from './init';
4
5  const first = new FirstType();
6
7  const boundSecond = bind(first, SecondType);
8
9  // { first: undefined, second: string }
10 const second = boundSecond();
11 console.log(second);
12
```



TS init.ts

TS apply.ts

TS call.ts

TS bind.ts

×

2023-11-HolyJS > examples > TS bind.ts > ...

```
1 import { bind } from 'mnemonica';
```

```
2
```

```
3 import { FirstType, SecondType } from './init';
```

```
4
```

```
5 const second: {  
6   [x: string]: new () => unknown;
```

```
7   first: undefined; ;
```

```
8   second: string;
```

```
9 // { f }
```

```
10 const second = boundSecond();
```

```
11 console.log(second);
```

```
12
```

TS init.ts

TS apply.ts

TS call.ts

TS bind.ts

×

2023-11-HolyJS > examples > TS bind.ts > ...

```
1 import { bind } from 'mnemonica';
```

```
2
```

```
3 import { FirstType, SecondType } from './init';
```

```
4
```

```
5 const second: {  
6   [x: string]: new () => unknown;
```

```
7   first: undefined;
```

```
8   second: string;
```

```
9 }  
// { f
```

```
10 const second = boundSecond();
```

```
11 console.log(second);
```

```
12
```



**... class ...**



TS init.ts

×

TS direct.ts

TS apply.ts

TS call.ts

TS bind.ts


2023-11-HolyJS > examples > TS init.ts > ...

```
1  import { define } from 'mnemonica';
2
3  export const FirstType = define('SomeType', function(this: {
4  |   first: 'FirstType',
5  | }) {
6  |   this.first = 'FirstType';
7  | });
8
9  export const SecondType = FirstType.define('SecondType', function(this: {
10 |   first: undefined,
11 |   second: string,
12 | }) {
13 |   this.first = undefined;
14 |   this.second = 'SecondType';
15 | });
16
```



```
9  export const SecondType = FirstType.define('SecondType',
10  →  function (
11  →  →  this: {
12  →  →  →  first: undefined,
13  →  →  →  second: string,
14  →  →  →  }) {
15  →  →  this.first = undefined;
16  →  →  this.second = 'SecondType';
17  →  });
18
19
20  export const ThirdType = SecondType.define('ThirdType', class {
21  →  third: string
22  →  constructor() {
23  →  →  this.third = 'ThirdType';
24  →  }
25  →  });
26
```

```
9 export const SecondType = FirstType.define('SecondType',
10   function (
11     → this: {
12     →   → first: undefined,
13     →   → second: string,
14     →   → }) {
15     →   this.first = undefined;
16     →   this.second = 'SecondType';
17   → });
18
19
20 export const ThirdType = SecondType.define('ThirdType', class {
21   → third: string
22   → constructor() {
23   →   → this.third = 'ThirdType';
24   →   → }
25   → });
26
```





TS init.ts

TS third.ts

×

TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS third.ts > ...

```
1 import { apply } from 'mnemonica';
```

```
2
```

```
3 import { FirstType, SecondType, ThirdType } from './init';
```

```
4
```

```
5 const first = new FirstType();
```

```
6
```

```
7 // { first: undefined, second: string }
```

```
8 const second = apply(first, SecondType);
```

```
9 console.log(second);
```

```
10
```

```
11 const third = apply(second, ThirdType);
```

```
12 console.log(third);
```

```
13
```



TS init.ts

TS third.ts

×

TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS third.ts > ...

```
1 import { apply } from 'mnemonica';
```

```
2
```

```
3 import { FirstType, SecondType, ThirdType } from './init';
```

```
4
```

```
5 const third: {  
6   [x: string]: new () => unknown;  
7   first: undefined;  
8   second: string;  
9   third: string;  
10 }
```

```
10
```

```
11 const third = apply(second, ThirdType);
```

```
12 console.log(third);
```

```
13
```



TS init.ts

TS third.ts



TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS third.ts > ...

```
1 import { apply } from 'mnemonica';
```

```
2
```

```
3 import { FirstType, SecondType, ThirdType } from './init';
```

```
4
```

```
5 const third: {  
6   [x: string]: new () => unknown;  
7   first: undefined;  
8   second: string;  
9   third: string;  
10 }
```

```
10
```

```
11 const third = apply(second, ThirdType);
```

```
12 console.log(third);
```

```
13
```





**пример**



**пример**





- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- как создаётся код для решения
- как развивается решение: что сделано
- **практический пример использования**

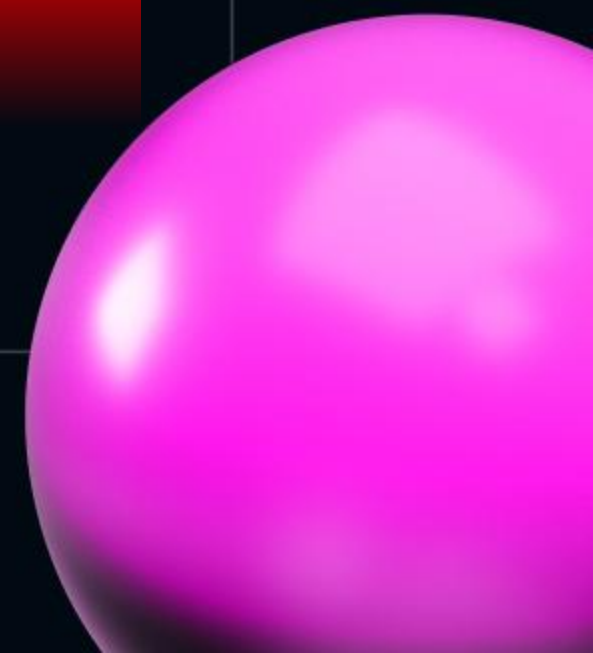


- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- как создаётся код для решения
- как развивается решение: что сделано
- **практический пример использования**





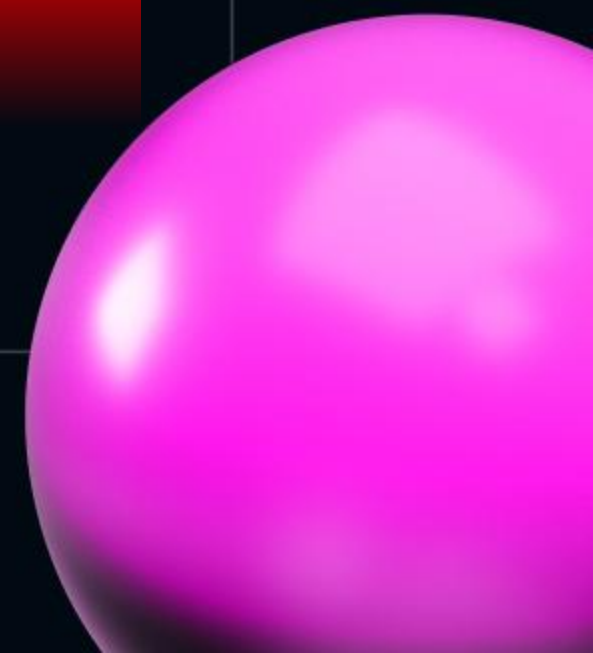
# Fata Morgana







**DEMO**







**ЧТО ДАЛЬШЕ**

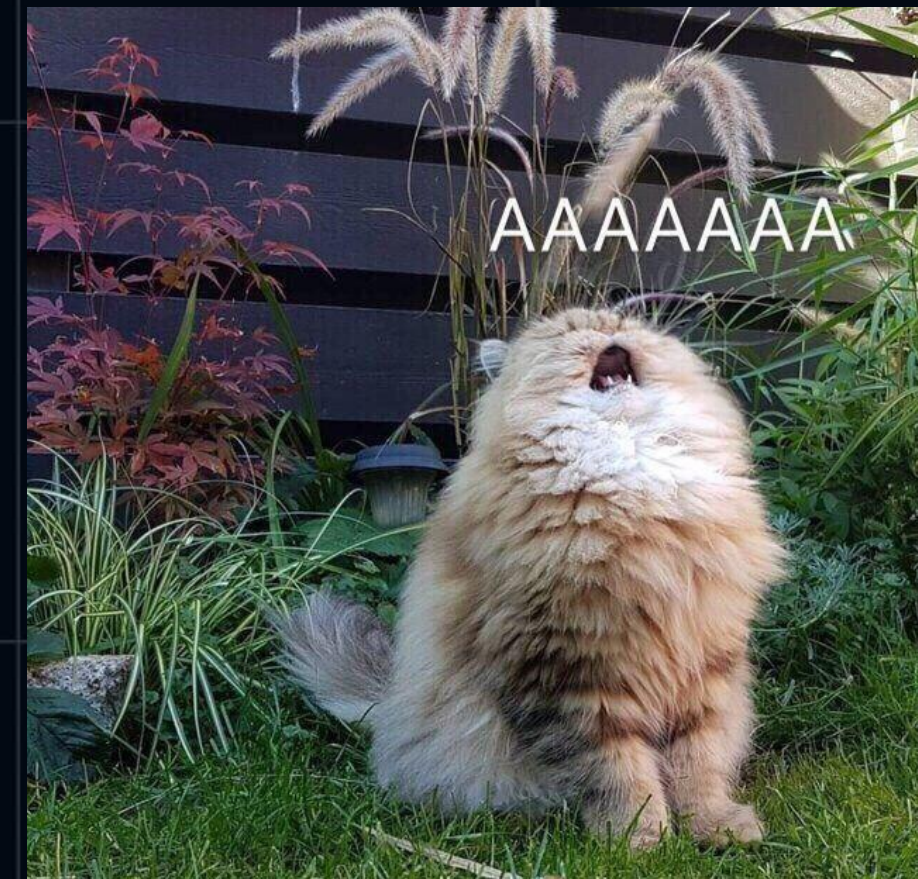


**ЧТО ДАЛЬШЕ**





- контекст постановки задачи
- формулировка проблематики
- про что уже рассказывал по теме
- как создаётся код для решения
- как развивается решение: что сделано
- практический пример использования
- **что ещё можно сделать дальше**



**... decorators ...**





TS init.ts

TS decorator.ts X

TS third.ts

TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS decorator.ts > ...

```
1  import { define } from 'mnemonica';
2
3  function defined <T extends { new (): unknown }> (cstr: T, s: ClassDecoratorContext<T>) {
4  |→   const TypeDef = define(s.name, cstr);
5  |→   Object.setPrototypeOf(cstr.prototype, new TypeDef);
6  |}
7
8  @defined
9  class MyClass {
10 |→   z: number;
11 |→   constructor () {
12 |→     |→   this.z = 123;
13 |→   }
14 |}
15
16  const myInstance = new MyClass;
17  console.log(myInstance.z);
18
```

TS init.ts

TS decorator.ts X

TS third.ts


TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS decorator.ts > ...

```
1 import { define } from 'mnemonica';
2
3 function defined <T extends { new (): unknown }> (cstr: T, s: ClassDecoratorContext<T>) {
4   const TypeDef = define(s.name, cstr);
5   Object.setPrototypeOf(cstr.prototype, new TypeDef);
6 }
7
8 @defined
9 class MyClass {
10   z: number;
11   constructor () {
12     this.z = 123;
13   }
14 }
15
16 const myInstance = new (property) MyClass.z: number
17 console.log(myInstance.z);
18
19
```





... HO ...



TS init.ts

TS decorator.ts 1 X

TS third.ts

TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS decorator.ts > ...

```
1 import { define } from 'mnemonica';
2
3 function defined <T extends { new (): unknown }> (cstr: T, s: ClassDecoratorContext<T>) {
4   const TypeDef = define(s.name, cstr);
5   Object.setPrototypeOf(cstr.prototype, new TypeDef);
6 }
7
8 @defined
9 class MyClass {
10   z: number;
11   constructor () {
12     this.z = 123;
13   }
14 }
15
16 const myInstance = new MyClass;
17 console. Property 'define' does not exist on type 'typeof MyClass'. ts(2339)
18
19 any
20 View Problem (Ctrl+K N) Quick Fix... (Ctrl+.)
21 MyClass.define()
```



TS init.ts

TS decorator.ts 1 X

TS third.ts

TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS decorator.ts > ...

```
1 import { define } from 'mnemonica';
2
3 function defined <T extends { new (): unknown }> (cstr: T, s: ClassDecoratorContext<T>) {
4   const TypeDef = define(s.name, cstr);
5   Object.setPrototypeOf(cstr.prototype, new TypeDef);
6 }
7
8 @defined
9 class MyClass {
10   z: number;
11   constructor () {
12     this.z = 123;
13   }
14 }
```

```
16 const myInstance = new MyClass;
```

```
17 console.log(myInstance);
```

Property 'define' does not exist on type 'typeof MyClass'. ts(2339)

any

[View Problem \(Ctrl+K N\)](#) [Quick Fix... \(Ctrl+.\)](#)

```
21 MyClass.define()
```



и ещё на самом деле  
`myInstance` **ХОТЬ И** как бы  
является экземпляром того,  
что создаётся от `define`,  
ТОЛЬКО ВОТ ...





TS init.ts

TS decorator.ts 1 X

TS third.ts

TS apply.ts

TS call.ts

TS bind.ts

2023-11-HolyJS > examples > TS decorator.ts > ...

```
1 import { define } from 'mnemonica';
2
3 function defined <T extends { new (): unknown }> (cstr: T, s: ClassDecoratorContext<T>) {
4   const TypeDef = define(s.name, cstr);
5   Object.setPrototypeOf(cstr.prototype, new TypeDef);
6 }
```

```
7
8 @defined
9 class MyClass {
10   z: number;
11   constructor () {
12     this.z = 123;
13   }
14 }
```

```
15
16 const myInstance = new MyClass;
```

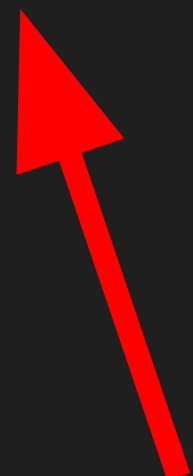
```
17 console.log(myInstance);
```

Property 'define' does not exist on type 'typeof MyClass'. ts(2339)

any

[View Problem \(Ctrl+K N\)](#) [Quick Fix... \(Ctrl+.\)](#)

```
21 MyClass.define()
```



**... ТАК ЧТО ...**







**Спасибо !**







следующий доклад

# Mnemonic Project



**Виктор  
Вершанский**

