

БЭМ в дикой природе

.block__el__el

.b-block

.block--mod

...

Igor Zenich

Frontend developer



delaz

38 (050) 401 81 39

delka.name

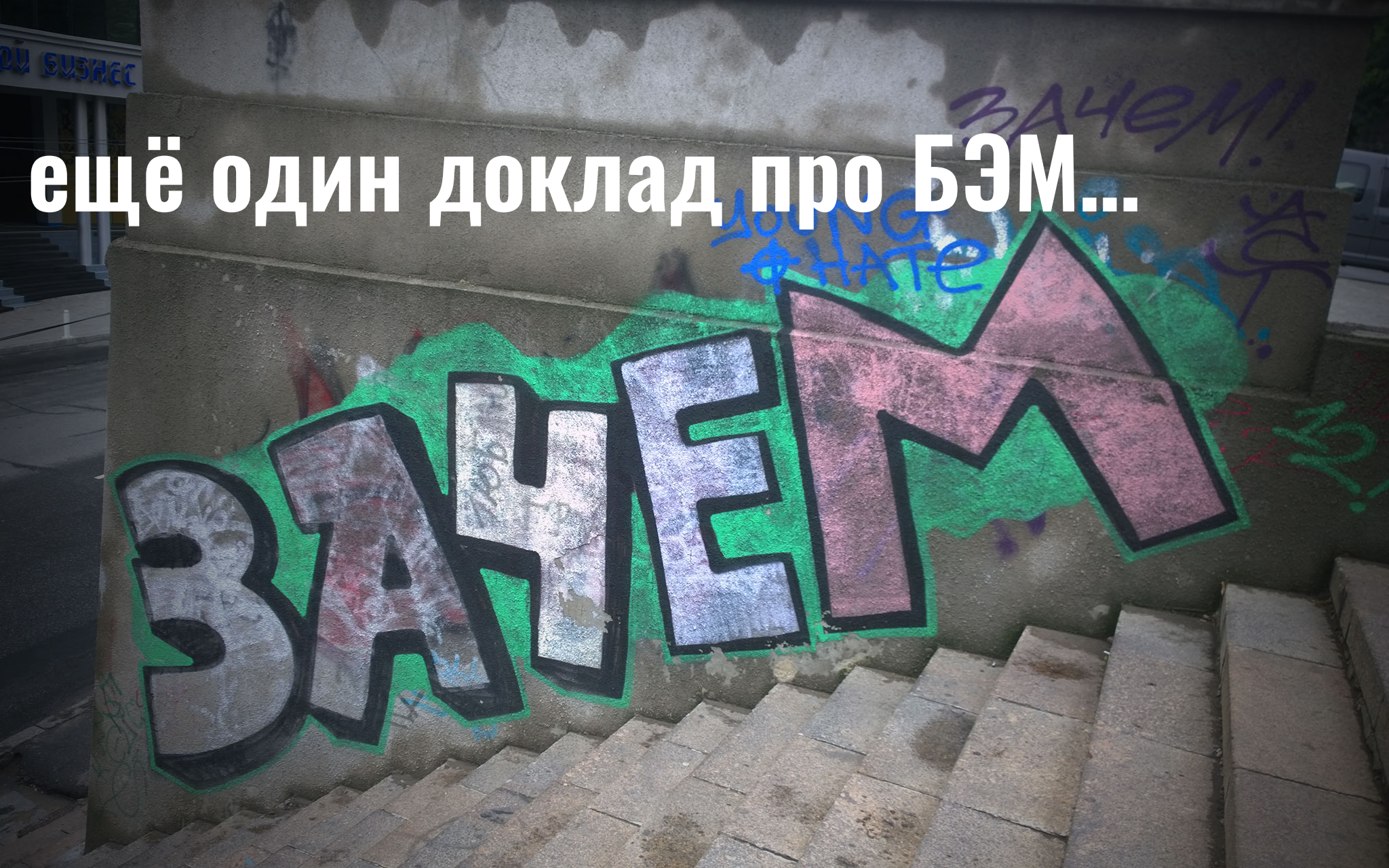


В исходниках этой презентации спрятались подсказки!

В комментах html написаны тезисы доклада и комментарии к слайдам требующим пояснений.

Презентация: bit.ly/bem-in-the-wild

bit_lv/hem-in-
the-wild



ещё один доклад про БЭМ...

В прошлом году мы обсуждали ручную вёрстку по БЭМ...

Презентация:

bit.ly/bem-css

Видео доклада:



[watch?v=kBgHdSOj33A](https://www.youtube.com/watch?v=kBgHdSOj33A)

Презентация содержит больше слайдов и примеров, чем оригинальное видео.

“

— Мы используем БЭМ, но не *FullStack*, а *#b_в CSS*.

— Нет, не как в Яндексе.

— Как?

— Ну как Гарри Робертс писал...

— Как именно, он разное писал?

— *Fuck*.



Andrey Melikhov

@amel_true



Даже внутри специализированной рассылки слова
"они перешли на БЭМ!" могут означать что угодно

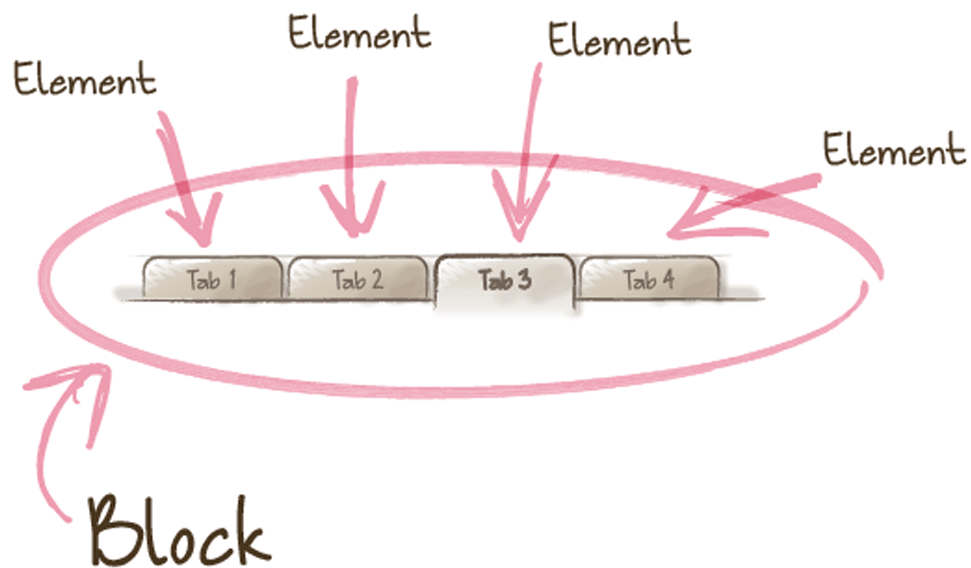
2:45 PM - 20 Nov 2015

1 RETWEET 1 FAVORITE



Each time we have to understand what exactly BEM we are talking about. Both with customer and internally in the team.

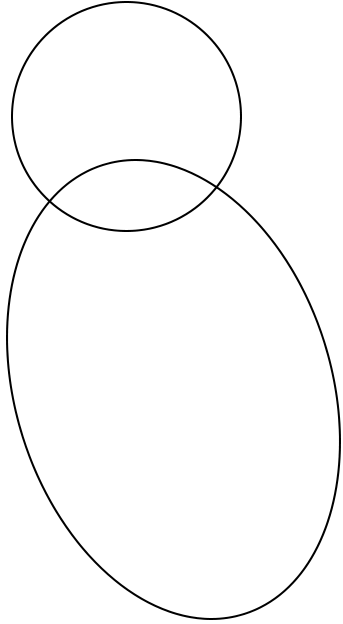
**Непонятость
БЭМ**



```
$ git clone git://github.com/bem/project-stub.git
start-pretty-project
$ cd start-pretty-project/
$ rm -rf .git
$ git init
$ npm install
$ sudo npm install bem-cli -g
$ bem make
$ bem server
content: [
  {
    block: 'head',
    content: {
      block: 'layout',
      content: [
        {
          elem: 'left',
          content: 'left here'
        },
        {
          elem: 'right',
```

В основе нашей методологии лежат блоки, которые могут использоваться, как сами по себе, так и вместе с другими блоками. Блоки состоят из элементов.

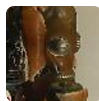
HTML-разметка web-страницы и применяемые к ней CSS-правила генерируются из ее описания в BEMJSON-файле `pageName.bemjson.js`. В терминах БЭМ будем называть BEMJSON-описание страницы декларацией. BEMJSON-декларация - это структура страницы, описанная в терминах блоков, элементов и модификаторов...



1. Draw some circles



2. Draw the rest of fucking owl

**Разработчик**

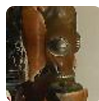
@jsunderhood



БЭМ пугал когда он вышел. Там была простыня текста в документации на не очень прямом английском про философию.

10:06 - 6 августа 2015

1 ТВИТ В ИЗБРАННОМ

**Разработчик**

@jsunderhood



Что я имел ввиду - авторы БЭМ не смогли его продать. В том числе из-за сложной и не очень удобной документации, ИМХО.

10:06 - 6 августа 2015

1 РЕТВИТОВ 4 ТВИТОВ В ИЗБРАННОМ

 ProTip! Type `g` and then `i` on any issue or pull request to go back to this listing page.

Менялись инструменты.

**Методология не
менялась**

Диалекты БЭМ



Vitaly Harisov

@harisov



БЭМ это фреймворк для создания фреймворков.

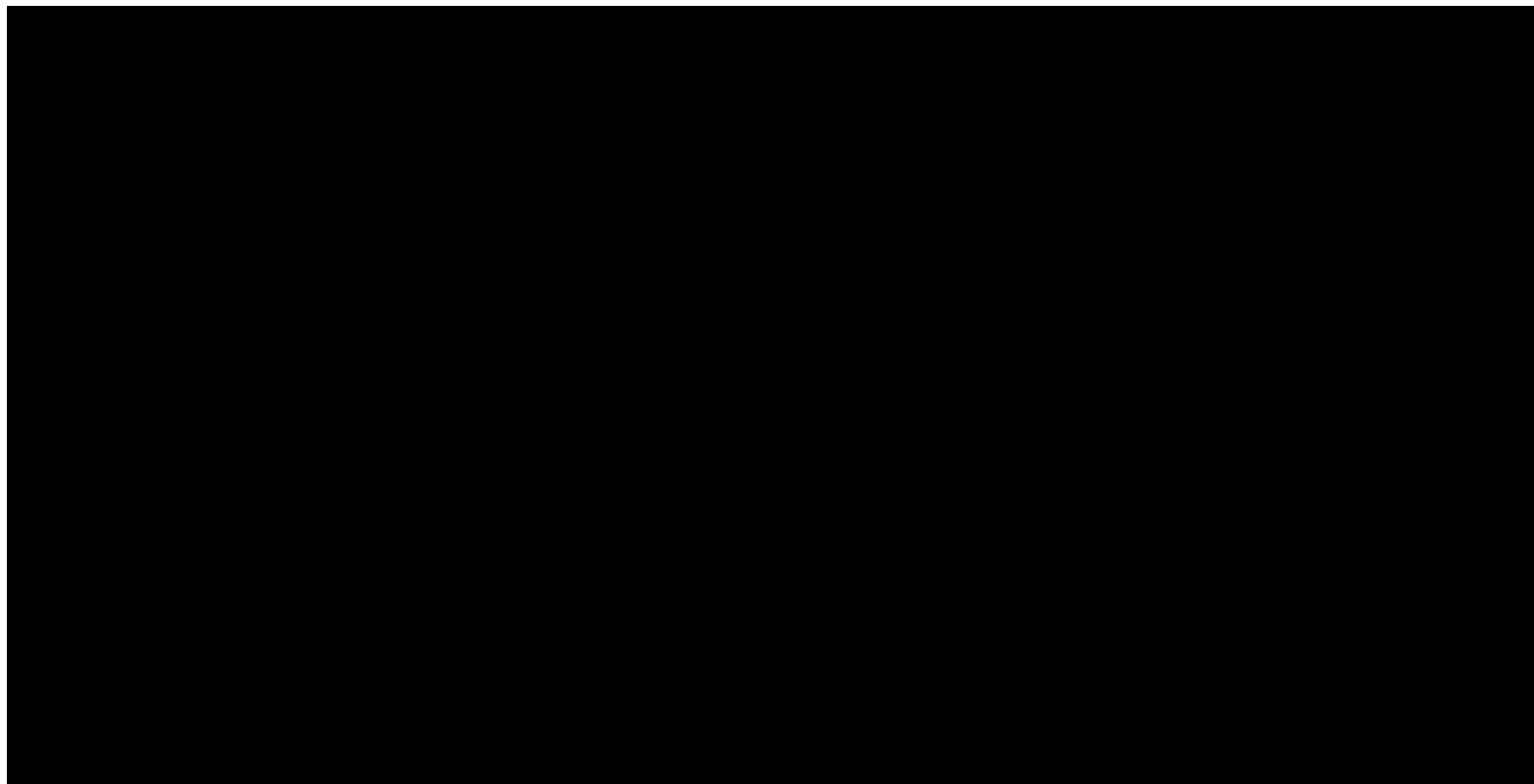
14:50 - 12 июля 2015 Ukraine, Ukraine

3 РЕТВИТОВ 1 ТВИТОВ В ИЗБРАННОМ

Вот это вот всё на 10 минут:

- Префиксы (b-)
- Пространства имен (BEMIT)
- Стилль Гарри Робертса, camelCase и «без подчеркиваний»
- Сокращённые модификаторы (стилль „No-namespace“)
- JS-блоки
- OPOR
- BEViS
- Google MDL
- BEM project-stub HTML edition

Префиксы



Префиксы

Были в раннем БЭМ. Сейчас пропагандируются Гарри Робертсом и используются многими не-Яндекс разработчиками. Используются для создания своего пространства имен и логического разделения блоков.

ВЕМІТ: Пространства имен

Продвинутое использование префиксов и суффиксов от Гарри Робертса. Попытка описать взаимосвязь между независимыми блоками с точки зрения SMACSS и OOCSS.

Стиль Гарри Робертса

Многим нравится зарубежный формат модификаторов, через „--“, он читабельней.

```
<a class="block-name__element-name --state_active">
```

Стиль camelCase

А через camelCase - ещё читабельней!

```
<a class="blockName__elementName --state_active">
```

Стиль без подчеркиваний

Некоторые идут ещё дальше и заменяют „__“ на „-“. camelCase единственный гарантирует что вы поймете где блок, а где элемент.

```
<a class="blockName-elementName --state_active">
```


Сокращенные модификаторы

Сокращенные модификаторы

Правильно писать так:

```
.block-name__elem_key_value
```

Или так:

```
.blockName__elem --key_value
```

Но такие имена классов тяжело читать

```
<div class="block-name block-name_key1_val1 block-name_key2_val2 block-name_key3_val3 ">
```

Нет дуракоустойчивости, модификатор могут перенести без блока:

```
<div class="block-name_key1_val1 some-another-block">
```

Хочется так:

Сокращенные модификаторы

Зарубежом их называли „**Individual modifiers: a shorter syntax**“. У нас перевели как «**Обособленные модификаторы: сокращенный синтаксис**». Яндекс в официальной документации называет их «**Стиль No-namespace**».

```
01. <div class="blockName__elem -key_value ">
02. .blockName {
03.   &__elem {
04.     &.-key_value {
05.     }
06. }
```



WARNING

- Теряем миксы блоков с модификаторами!
- Не можем использовать Full BEM Stack!

Решением может быть запись модификаторов в столбик

```
<div class="block-name block-name_key1_value1  
block-name_key2_value2 block-name_key3_value3 ">
```

01. <!-- VS -->

02. <div class="block-name

03. block-name_key1_value1

04. block-name_key2_value2

05. block-name_key3_value3 ">

А ещё можно юзать Jade, а в Яндексе вообще не пишут html-код руками, у них BEMJSON.

JS-блоки

```
$( ' .js-fancybox' ).fancybox();
```

Это миксование css-блока и js-блока на одной dom-ноде.

Канонический БЭМ считает, что они не нужны, т.к. js-функционал нет смысла отделять от блока. Гарри Робертс и не-Яндекс разработчики их активно используют и пропагандируют: т.к. разделение позволяет легко копировать css-блок без связанного с ним JS.

Альтернативные реализации: OPOR

Известный пример использования методологии БЭМ сторонним разработчиком - Артёмом Сапегиним

Альтернативные реализации: BEViS

Диалект БЭМ, с более строгими правилами для максимальной надежности верстки, придуманный Вадимом Макишвили для Яндекс.Карт.

Альтернативные реализации: Google MDL

Библиотека блоков от Google, css-реализации их Material Design придуманного для Android.

Full BEM Stack в HTML

Вы можете создавать свои гайдлайны

БЭМ дает лишь базовый набор правил, конкретную реализацию и синтаксис вы выбираете сами.

- [Harry Roberts](#)
- [iDeus](#)
- [Artem Sapegin](#)
- [CodeRiver](#) (Yuriy Artyukh, cssing.org.ua)
- [\[redacted\]](#)

Это всё БЭМ

Тонкости БЭМ

БЭМ включает в себя много техник

Кое-что мы с вами обсудили/упомянули:

- почему нельзя вкладывать блоки
- [bem-tools](#) и альтернативные тулзы для упрощенного написания

БЭМ-нейминга в HTML:

- [BEML](#) — html-препроцессор для BEM
- [posthtml-bem](#) — тоже самое, но через PostHTML
- [bemto](#) — миксины для написания BEM в Jade
- [миксование](#)
- зачем нужны пространства имен/префиксы (включая `js-` блоки)

Но многое осталось за кадром

- Я не рассказал вам как разбивать блоки на файлы и складывать в файловой системе
- почему не нужно использовать не использовать `@extend` в Sass
- о `i-bem.js`
- и об абстрактных блоках `(i-)`
- ...контекстных блоках `(m-)`
- ...холдер-блоках `(h-)`
- ...уровнях переопределения
- ...глобальных модификаторах
- ...и про историю ВЕМ — почему все так?

Читать дальше

- [Советы от ведущих БЭМ-разработчиков: ошибки и best practices](#)
- [BEM Quick Start Guide](#) — БЭМ: UnOfficial Team
- [Мажорный релиз новой документации](#) — БЭМ: Яндекс Team
- [\[redacted\]](#) — Владимир Гриненко
- [Классификация ошибок BEM CSS](#) (со стр. 50) — Александр Корецкий [@n2j7](#)
- [Material Design Lite CSS](#) — Google
- [\[redacted\]](#)
- [варианты использования префиксов от Гарри Робертса](#)
- [BEMIT](#) — Harry Roberts
- [\[redacted\]](#) и [\[redacted\]](#) — Вадим Макишвили

...и дальше

- что такое блок, префикс и что такое независимый блок
- контекстные блоки (темы) (раздел «Внутри контекстного блока»)
- layout-блоки и холдер-блоки
- как позиционировать БЭМ-блоки относительно друг друга
- абстрактные блоки (раздел „i-, от include“) и их реализация в Sass через или лучше через
- (раздел «Модификация файлами») и
-
-
- Откуда взялся зарубежный синтаксис модификаторов и is-

Спасибо!

Igor Zenich

EPAM



- [delka.name](#)
- [Redacted]
- [igor@delka.name](#)

Презентация: [bit.ly/bem-in-the-wild](#)