



Экзотическая шаблонизация

или как писать HTML для блоков

Сергей Бережной
руководитель отдела
разработки поисковых интерфейсов

Вступление



Вступление

Кто я?



Кто я?

- Разработчик интерфейсов Яндекса с 2005 года
- Руководитель отдела разработки поисковых интерфейсов
- БЭМ-соавтор

 github.com/veged

 [@veged](https://twitter.com/veged)



Вступление

КТО ВЫ?



Основная часть



Основная часть

Классика



Классика

[Погода](#)  +4 °C

[Пробки](#)  2 балла ↓

Классика

```
{
  settings: {
    weather: true,
    traffic: false
  },
  weather: { temp: 4 },
  traffic: { level: 2 }
}
```

Классика

```
<ul>  
  [% IF settings.weather %]  
    <li>[% weather.temp %]</li>  
  [% END %]  
  [% IF settings.traffic %]  
    <li>[% traffic.level %]</li>  
  [% END %]  
</ul>
```

Классика

Data → HTML

Основная часть

Альтернатива

Альтернатива

```
{  
  settings: {  
    weather: true,  
    traffic: false  
  },  
  weather: { temp: 4 },  
  traffic: { level: 2 }  
}
```

Альтернатива

```
var items = [];  
  
if(settings.weather) items.push({  
  elem: 'item',  
  content: weather.temp });  
  
if(settings.traffic) items.push({  
  elem: 'item',  
  content: traffic.level });  
  
return { block: 'menu', content: items };
```

Альтернатива

```
{  
  block: 'menu',  
  content: [  
    {  
      elem: 'item',  
      content: 4  
    }  
  ]  
}
```

Альтернатива

```
block menu {  
  tag: 'ul'  
  elem item, tag: 'li'  
}
```


Альтернатива

```
<ul class="menu">  
  <li class="menu__item">  
    +4 °C  
  </li>  
</ul>
```

Альтернатива

Data → View → HTML

Альтернатива

Data → БЭМ-дерево → HTML

Альтернатива

Data → БЭМ-дерево —(BEMHTML)— HTML

Альтернатива

- блоки рисуются сами



Альтернатива

- блоки рисуются сами
- не нужна интеграция нового HTML в шаблоны

Альтернатива

```
<ul>
  [% IF settings.weather %]
    <li>[% weather.temp %]</li>
  [% END %]
  [% IF settings.traffic %]
    <li>[% traffic.level %]</li>
  [% END %]
</ul>
```

Альтернатива

- блоки рисуются сами
- не нужна интеграция нового HTML в шаблоны
- можно прототипировать HTML/CSS без реальных данных

Альтернатива

Data → БЭМ-дерево → HTML

Альтернатива

~~Data~~ → БЭМ-дерево → HTML

Альтернатива

- блоки рисуются сами
- не нужна интеграция нового HTML в шаблоны
- можно прототипировать HTML/CSS без реальных данных
- меньше копипаста, больше повторного использования

Альтернатива

```
<ul>
  [% IF settings.weather %]
    <li>[% weather.temp %]</li>
  [% END %]
  [% IF settings.traffic %]
    <li>[% traffic.level %]</li>
  [% END %]
</ul>
```

Альтернатива

```
block menu {  
  tag: 'ul'  
  elem item, tag: 'li'  
}
```

Альтернатива

- блоки рисуются сами
- не нужна интеграция нового HTML в шаблоны
- можно прототипировать HTML/CSS без реальных данных
- меньше копипаста, больше повторного использования
- очень похоже на CSS

Альтернатива

```
block menu {  
  tag: 'ul'  
  elem item, tag: 'li'  
}
```

Альтернатива

```
/* CSS */  
.header {  
  font-size: 140%;  
}
```

```
// BEMHTML  
block header, tag: 'h1'
```


Основная часть

Файлы

Файлы

blocks/

Файлы

blocks/
header/
menu/

Файлы

```
blocks/  
  header/  
  menu/  
    __item/
```

Файлы

```
blocks/  
  header/  
    header.css  
  menu/  
    __item/  
      menu__item.css  
    menu.css
```

Файлы

blocks/

header/

header.css

header.bemhtml

menu/

__item/

menu__item.css

menu__item.bemhtml

menu.css

menu.bemhtml

Файлы

blocks/
pages/

Файлы

blocks/
pages/
index/

Файлы

```
blocks/  
pages/  
  index/  
    index.css
```

Файлы

pages/index/index.css

```
@import url(.././blocks/header/header.css);  
@import url(.././blocks/menu/menu.css);  
@import url(.././blocks/menu/__item/menu__item.css);
```

Файлы

```
bem build
```

```
--output-name=index
```

```
--decl=pages/index/index.bemdecl.js
```

```
--level=blocks/
```

```
--tech=css
```

Файлы

```
blocks/  
pages/  
  index/  
    index.bemdecl.js // декларация
```

Файлы

```
bem build
```

```
--output-name=index
```

```
--decl=pages/index/index.bemdecl.js
```

```
--level=blocks/
```

```
--tech=css
```

```
--tech=bemhtml
```

Файлы

```
blocks/  
pages/  
  index/  
    index.bemdecl.js // декларация
```

Файлы

blocks/

pages/

index/

index.bemdecl.js // декларация

index.css // сборка

index.bemhtml.js // сборка

Файлы

bem.info/tools/bem

Основная часть

Переопределения

Переопределения

```
/* CSS */  
.header {  
    font-size: 140%;  
    font-weight: bold;  
}
```

Переопределения

```
<!-- HTML -->  
<h1 class="header">  
    Заголовок 1  
</h1>
```

Переопределения

```
// БЕМHTML  
block header, tag: 'h1'
```

Переопределения

```
blocks/  
  header/  
    header.css  
    header.bemhtml
```

Переопределения

```
/* CSS */  
  
.header {  
    font-size: 140%;  
    font-weight: bold;  
}  
  
.header_level_2 {  
    font-size: 120%;  
}
```

Переопределения

```
blocks/  
  header/  
    _level/  
      header_level_2.css  
header.css
```

Переопределения

```
<!-- HTML -->  
<h2 class="header header_level_2">  
    Заголовок 2  
</h2>
```


Переопределения

```
// БЕМHTML
```

```
block header, tag: 'h1'
```

```
block header, mod level 2, tag: 'h2'
```

Переопределения

```
blocks/  
  header/  
    _level/  
      header_level_2.css  
      header_level_2.bemhtml  
header.css  
header.bemhtml
```

Переопределения

pages/index/index.css

```
@import url(.././blocks/header/header.css);  
@import url(.././blocks/header/_level/header_level_2.css);
```

Переопределения

```
mylib/  
  blocks/  
    header/  
      _level/  
        header_level_2.css  
        header_level_2.bemhtml  
      header.css  
      header.bemhtml
```

Переопределения

```
mylib/  
  blocks/  
    header/  
      _level/  
        header_level_2.css  
        header_level_2.bemhtml  
      header.css  
      header.bemhtml  
  blocks/
```

Переопределения

```
/* CSS */  
.header {  
    font-weight: normal;  
}
```

Переопределения

```
<!-- HTML -->
```

```
<h2 class="header">
```

Заголовок 1

```
</h2>
```

```
<h3 class="header header_level_2">
```

Заголовок 2

```
</h3>
```

Переопределения

```
// БЕМHTML
```

```
block header, tag: 'h2'
```

```
block header, mod level 2, tag: 'h3'
```


Переопределения

```
mylib/  
blocks/  
  header/  
    _level/  
      header_level_2.bemhtml  
header.css  
header.bemhtml
```

Переопределения

```
bem build
```

```
  --output-name=index
```

```
  --decl=pages/index/index.bemdecl.js
```

```
  --level=blocks/
```

```
  --tech=css
```

```
  --tech=bemhtml
```

Переопределения

```
bem build
```

```
  --output-name=index
```

```
  --decl=pages/index/index.bemdecl.js
```

```
  --level=mylib/blocks/
```

```
  --level=blocks/
```

```
  --tech=css
```

```
  --tech=bemhtml
```

Переопределения

pages/index/index.css

```
@import url(../../mylib/blocks/header/header.css);  
@import url(../../blocks/header/header.css);
```

Основная часть

Многопроходность

Многопроходность

```
<ul class="menu">  
  <li class="menu__item">  
    +4 °C  
  </li>  
</ul>
```

Многопроходность

```
{  
  block: 'menu',  
  content: [  
    {  
      elem: 'item',  
      content: 4  
    }  
  ]  
}
```

Многопроходность

```
block menu {  
  tag: 'ul'  
  elem item, tag: 'li'  
}
```


Многопроходность

```
<ul class="menu">  
  <li class="menu__item">  
    <span class="menu__item-wrap">  
      +4 °C  
    </span>  
  </li>  
</ul>
```

Многопроходность

```
block menu {  
  tag: 'ul'  
  elem item, tag: 'li'  
  elem item-wrap, tag: 'span'  
}
```

Многопроходность

```
{
  block: 'menu',
  content: [
    {
      elem: 'item',
      content: {
        elem: 'item-wrap',
        content: 4
      }
    }
  ]
}
```

Многопроходность

```
block menu {
  tag: 'ul'
  elem item {
    tag: 'li'
    content: {
      elem: 'item-wrap',
      content: applyNext()
    }
  }
  elem item-wrap, tag: 'span'
}
```

Многопроходность

```
{  
  block: 'menu',  
  content: [  
    {  
      elem: 'item',  
      content: 4  
    }  
  ]  
}
```

Многопроходность

```
<ul class="menu">  
  <li class="menu__item">  
    <span class="menu__item-wrap">  
      +4 °C  
    </span>  
  </li>  
</ul>
```

Основная часть

JavaScript

JavaScript



JavaScript

[Личные настройки](#)

veged

[Почта](#)



[35932](#)



[Написать](#)



[Выход](#)

JavaScript

```
{  
  block: 'ya-user',  
  content: 'veged'  
}
```

JavaScript

```
<a  
  class="ya-user"  
  href="//i.yandex.ru">  
    <span class="ya-user__first">  
      v  
    </span>  
    eged  
</a>
```

JavaScript

```
block ya-user {  
  tag: 'a'  
  attrs: { href: '//i.yandex.ru' }  
  
  elem first, tag: 'span'  
}
```

JavaScript

```
{  
  block: 'ya-user',  
  content: [  
    { elem: 'first', content: 'v' },  
    'eged'  
  ]  
}
```

JavaScript

```
block ya-user, content: {  
  var oldContent = applyNext();  
  return [  
    {  
      elem: 'first',  
      content: oldContent[0]  
    },  
    oldContent.substring(1)  
  ]  
}
```

JavaScript

```
{  
  block: 'ya-user',  
  content: 'veged'  
}
```

JavaScript

```
<a  
  class="ya-user"  
  href="//i.yandex.ru">  
    <span class="ya-user__first">  
      v  
    </span>  
    eged  
</a>
```


Основная часть

Скорость

Скорость

github.com/veged/xjst

Secinājums

Secinājums

- декларативность

Secinājums

- декларативность
- блоки рисуются сами

Secinājums

- декларативность
- блоки рисуются сами
- переопределения

Secinājums

- декларативность
- блоки рисуются сами
- переопределения
- многопроходность

Secinājums

- декларативность
- блоки рисуются сами
- переопределения
- многопроходность
- БЭМ

Secinājums

- **существуют другие способы писать шаблоны**

Secinājums

- существуют другие способы писать шаблоны
- попробовать БЭМ: github.com/bem/project-stub

Secinājums

- существуют другие способы писать шаблоны
- попробовать БЭМ: github.com/bem/project-stub
- попробовать: XSLT, XJST, написать своё



Сергей Бережной

руководитель отдела
разработки
поисковых интерфейсов

veged@yandex-team.ru

 github.com/veged

 [@veged](https://twitter.com/veged)

Спасибо