

КОМПОНЕНТНЫЙ веб

Проникновение в дизайн

Антон Виноградов, Альфа-Лаборатория

FrontendConf 2015, 21-22 мая

Антон Виноградов

Разработчик интерфейсов

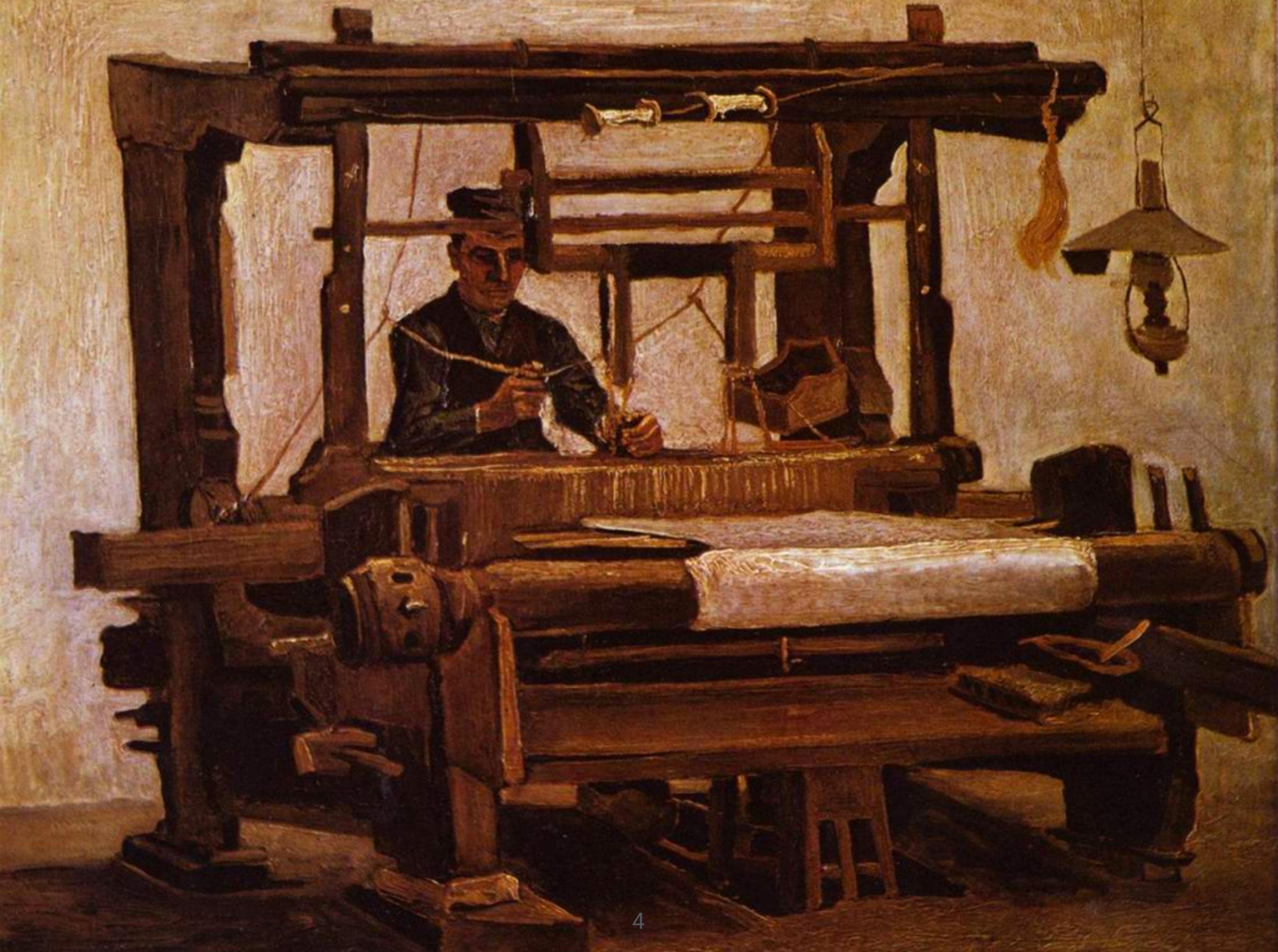
Альфа-Лаборатория

- более 3 лет занимаюсь фронтендом
- евангелист и БЭМ-контрибьютор
- автор OpenSource решений

@awinogradov













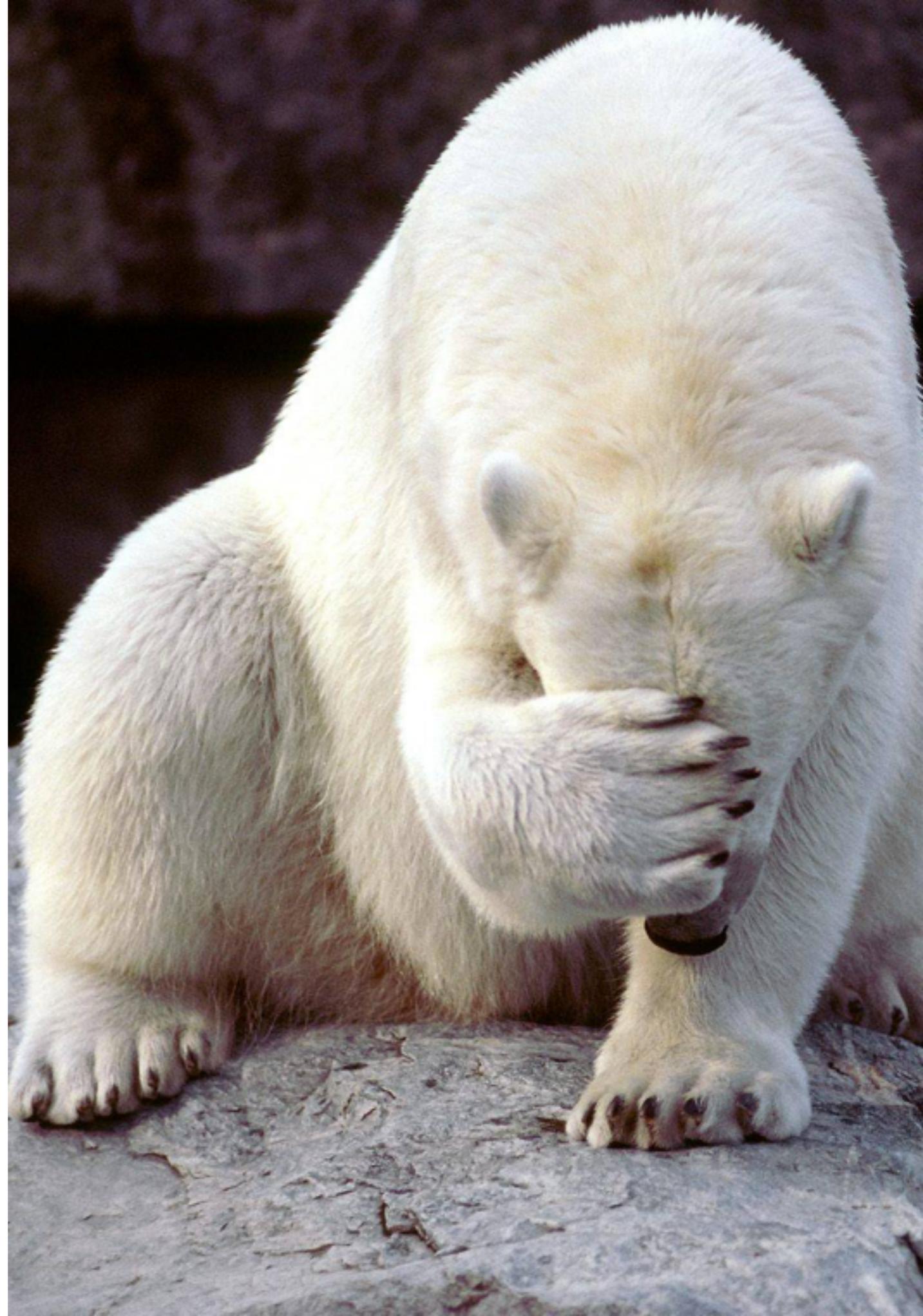
A



Везде одно и то же...

Везде одно и тоже

- **не** те подходы
- **не** те методологии
- и инструменты **не** те



Везде одно и то же ;-)

Оживляем UI Kit



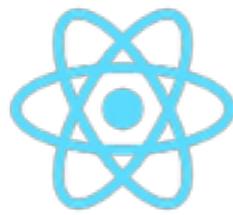
Оживляем UI Kit

- дизайнер в фотошопе



Оживляем UI Kit

- дизайнер в фотошопе
- разработчики во фреймворках



Оживляем UI Kit

- дизайнер в фотошопе
- разработчики во фреймворках
- КОМПОНЕНТНЫЙ ВЕБ



Оживляем UI Kit

- дизайнер в фотошопе
- разработчики во фреймворках
- компонентный веб
- генераторы гайдов



Оживляем UI Kit

- дизайнер в фотошопе
- разработчики во фреймворках
- компонентный веб
- генераторы гайдов
- дизайн в браузере



Оживляем UI Kit

- дизайнер в фотошопе
- разработчики во фреймворках
- компонентный веб
- генераторы гайдов
- дизайн в браузере
- учимся верстать



Все это про разработку...

Это не решает проблем взаимодействия
дизайнера и разработчика

Проблемы

- нет синхронизации
- разная методология
- 2 набора компонент
- изменения в одну сторону
- визуальное наследование
- верстка умерла

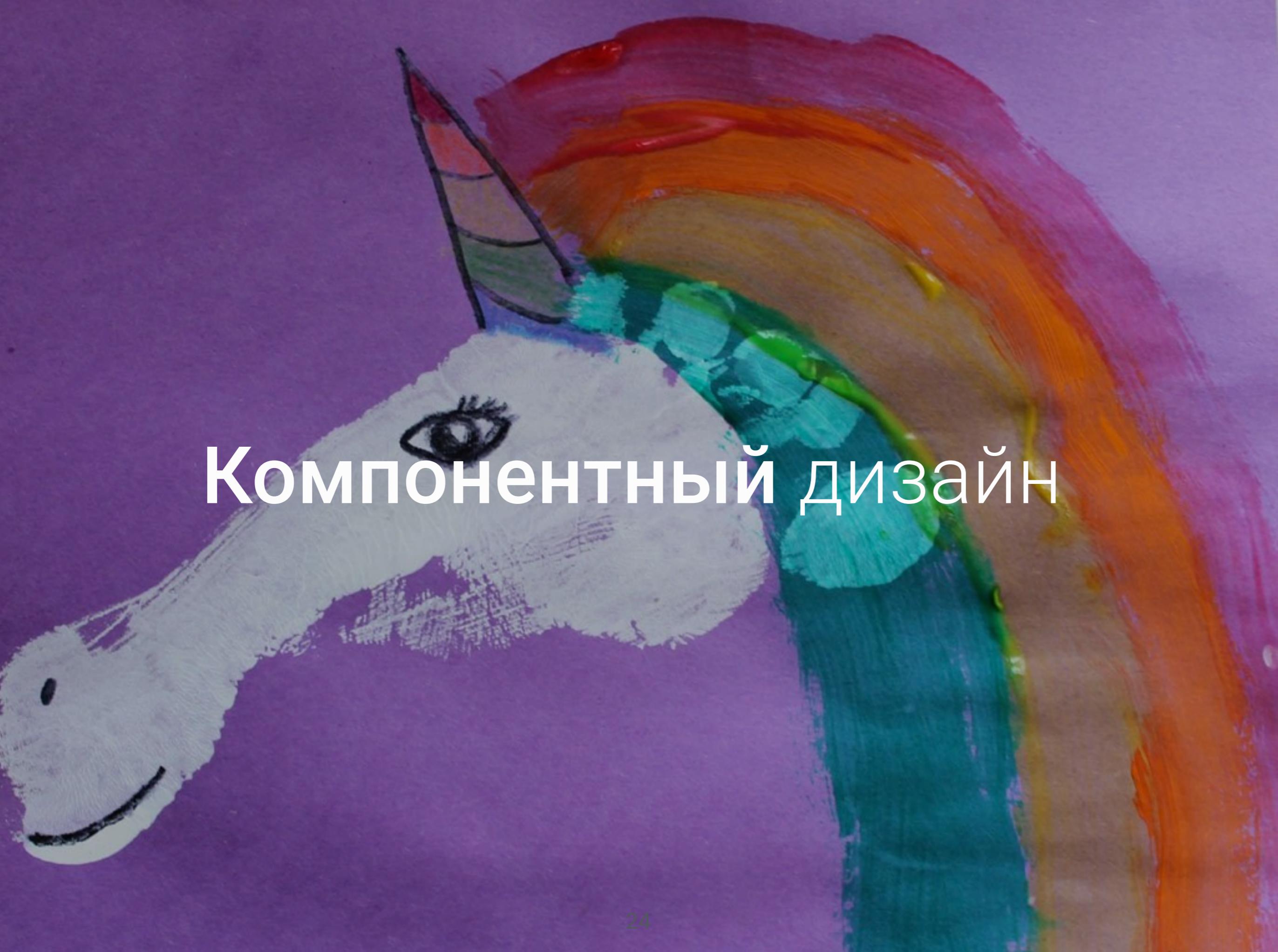


Дизайн в депрессии...

Решения

- **одна** методология на всех
- **один** набор компонент
- **один** принцип организации
- версионность **везде**
- **живые** компоненты первичны
- **наследование** на уровне реализации компонент
- **максимум** повторного использования



A child's drawing on a purple background. It depicts a unicorn with a white body and a large, bushy white mane. The unicorn's head is on the left, with a small black dot for an eye and a simple curved line for a smile. A black horn is drawn on its forehead. The mane and tail are rendered with thick, expressive brushstrokes in a rainbow spectrum of colors: red, orange, yellow, green, and blue. The text 'Компонентный дизайн' is overlaid in the center in a white, sans-serif font.

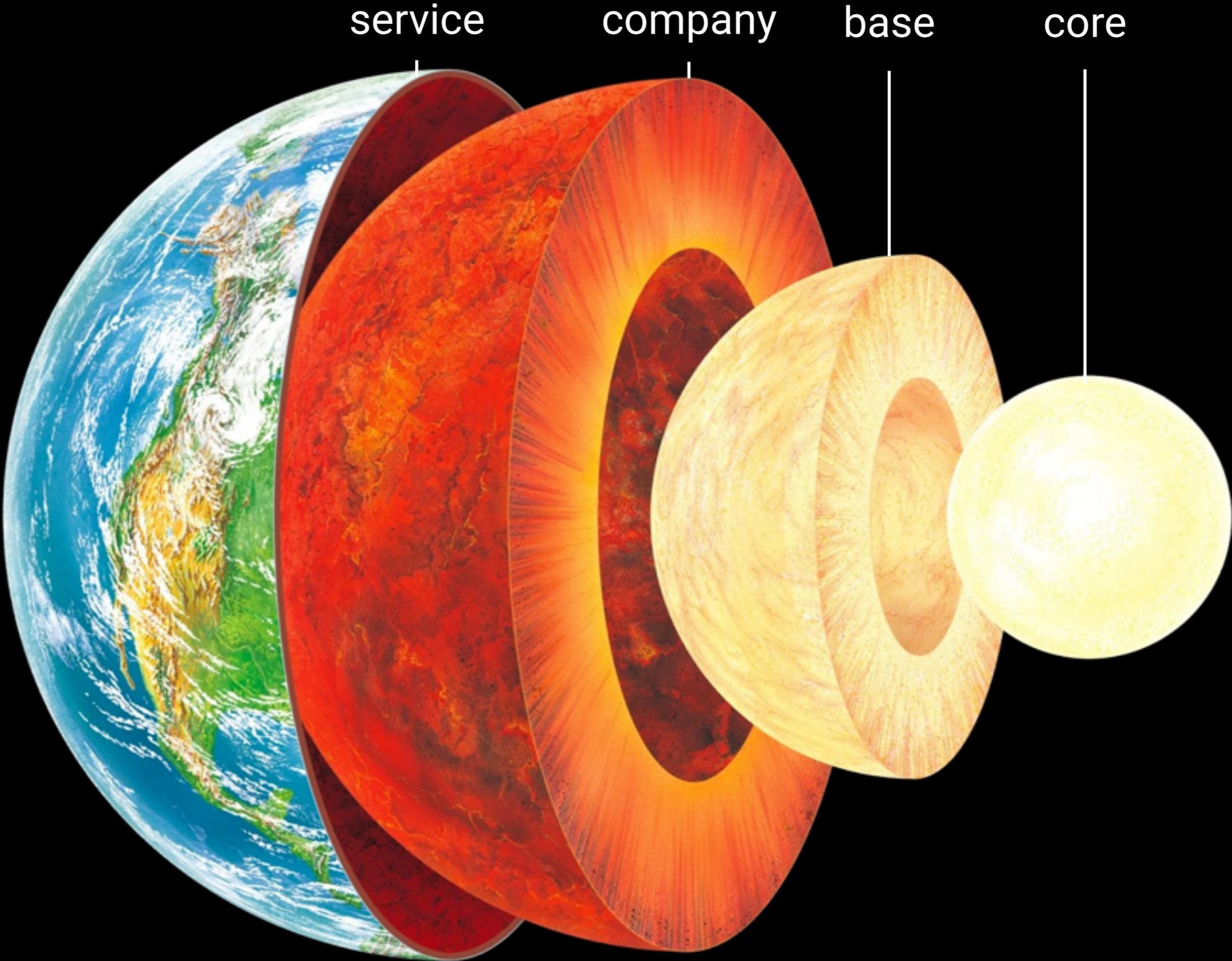
Компонентный дизайн



Дать **живые** компоненты
дизайнерам



**KEEP
CALM
AND
BEM**





```

<?xml version="1.0" encoding="utf-8"?>
<svg version="1.1" id="Layer_1" xmlns="http://www.w3.org/2000/svg"
  xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px"
  viewBox="-197 389.8 38 61.1" enable-background="new -197 389.8 38 61.1"
  xml:space="preserve">
<g>
<path fill="#fff" d="M-178,390.5c4,0,5.3,2.9,6.4,5.9c0.4,1.2,10.9,30.3,
  11.2,31.2c0.1,0.2,0,0.6-0.3,0.8c-0.4,0.1-5.8,2.3-6.1,2.4c-0.3,
  0.1-0.6-0.1-0.8-0.4c-0.1-0.3-2.4-6.6-2.6-7.3c-0.7,0-14.8,0-15.5,0c-0.3,
  0.7-2.5,7.1-2.6,7.3c-0.1,0.3-0.4,0.5-0.7,
  0.4c-0.3-0.1-5.8-2.3-6.1-2.4c-0.3-0.1-0.5-0.5-0.3-0.8c0.4-1.1,10.7-29.7,
  11.2-31.2C-183.3,393.4-182,390.5-178,390.5z M-196.4,442.5c-0.3,0-0.6,
  0.3-0.6,0.6v6.8c0,0.3,0.3,0.6,0.6,0.6h36.8c0.3,0,0.6-0.3,0.6-0.6
  v-6.8c0-0.3-0.3-0.6-0.6-0.6L-196.4,442.5L-196.4,442.5z M-178,
  400l-5.2,16h10.6L-178,400L-178,400z"/>
</g>
</svg>

```

```

<?xml version="1.0" encoding="utf-8"?>
<svg version="1.1" id="Layer_1" xmlns="http://www.w3.org/2000/svg"
  xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px"
  viewBox="-197 389.8 38 61.1" enable-background="new -197 389.8 38 61.1"
  xml:space="preserve">
<g>
<path fill="#fff" d="M-178,390.5c4,0,5.3,2.9,6.4,5.9c0.4,1.2,10.9,30.3,
  11.2,31.2c0.1,0.2,0,0.6-0.3,0.8c-0.4,0.1-5.8,2.3-6.1,2.4c-0.3,
  0.1-0.6-0.1-0.8-0.4c-0.1-0.3-2.4-6.6-2.6-7.3c-0.7,0-14.8,0-15.5,0c-0.3,
  0.7-2.5,7.1-2.6,7.3c-0.1,0.3-0.4,0.5-0.7,
  0.4c-0.3-0.1-5.8-2.3-6.1-2.4c-0.3-0.1-0.5-0.5-0.3-0.8c0.4-1.1,10.7-29.7,
  11.2-31.2C-183.3,393.4-182,390.5-178,390.5z M-196.4,442.5c-0.3,0-0.6,
  0.3-0.6,0.6v6.8c0,0.3,0.3,0.6,0.6,0.6h36.8c0.3,0,0.6-0.3,0.6-0.6
  v-6.8c0-0.3-0.3-0.6-0.6-0.6L-196.4,442.5L-196.4,442.5z M-178,
  400l-5.2,16h10.6L-178,400L-178,400z" />
</g>
</svg>

```



```

<?xml version="1.0" encoding="utf-8"?>
<svg version="1.1" id="Layer_1" xmlns="http://www.w3.org/2000/svg"
  xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px"
  viewBox="-197 389.8 38 61.1" enable-background="new -197 389.8 38 61.1"
  xml:space="preserve">
<g>
<path fill="#fff" d="M-178,390.5c4,0,5.3,2.9,6.4,5.9c0.4,1.2,10.9,30.3,
  11.2,31.2c0.1,0.2,0,0.6-0.3,0.8c-0.4,0.1-5.8,2.3-6.1,2.4c-0.3,
  0.1-0.6-0.1-0.8-0.4c-0.1-0.3-2.4-6.6-2.6-7.3c-0.7,0-14.8,0-15.5,0c-0.3,
  0.7-2.5,7.1-2.6,7.3c-0.1,0.3-0.4,0.5-0.7,
  0.4c-0.3-0.1-5.8-2.3-6.1-2.4c-0.3-0.1-0.5-0.5-0.3-0.8c0.4-1.1,10.7-29.7,
  11.2-31.2C-183.3,393.4-182,390.5-178,390.5z M-196.4,442.5c-0.3,0-0.6,
  0.3-0.6,0.6v6.8c0,0.3,0.3,0.6,0.6,0.6h36.8c0.3,0,0.6-0.3,0.6-0.6
  v-6.8c0-0.3-0.3-0.6-0.6-0.6L-196.4,442.5L-196.4,442.5z M-178,
  400l-5.2,16h10.6L-178,400L-178,400z"/>
</g>
</svg>

```





BEMHTML

bit.ly/bemhtml-tmpl

Кнопка

```
<button class="button">  
  <span class="button__text">  
    Кнопка  
  </span>  
</button>
```

BEMHTML

```
block('button')(
  tag('button'),

  content(function() {
    return {
      elem : 'text',
      content : applyNext()
    };
  }),

  elem('text').tag('span')
);
```

ReactJS

```
var Button = React.createClass({  
  render: function() {  
    return (  
      <button className='button'>  
        <span className='button__text'>  
          { this.state.content }  
        </span>  
      </button>  
    );  
  }  
});  
  
module.export = Button;
```

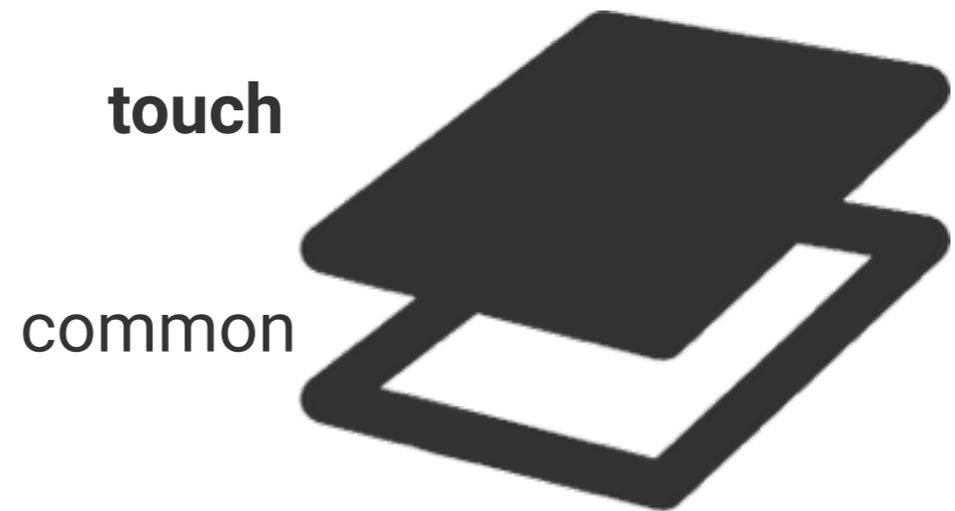
JSON

```
{  
  block : 'button',  
  content : 'Кнопка'  
}
```

HTML

```
<button class="button">  
  <span class="button__text">  
    Кнопка  
  </span>  
</button>
```

Схема наложения уровней кода





LEVEL UP!

Sketch +1



BEMHTML

```
block('button')(
  tag()('g'),

  content()(function() {
    return [
      { elem : 'instance' },
      applyNext()
    ];
  }),

  elem('instance').tag()('rect'),
  elem('text').tag()('text')
);
```

ReactJS

```
var Button = React.createClass({  
  render: function() {  
    return (  
      <g className='button'>  
        <rect className='button__instance'>  
          <text className='button__text'>  
            { this.state.content }  
          </text>  
        </g>  
      );  
    }  
  });  
  
module.export = Button;
```

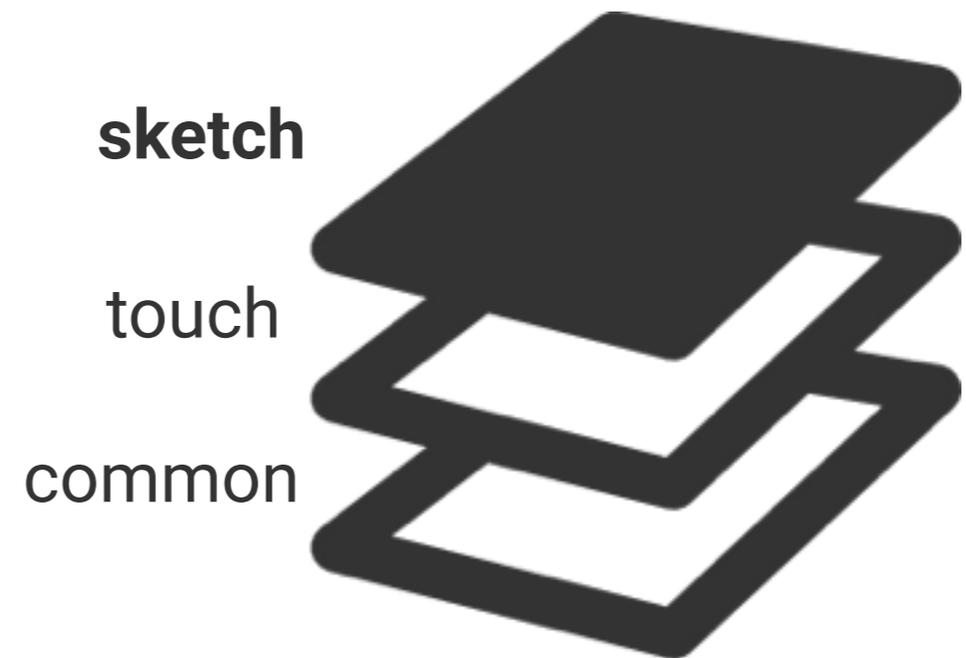
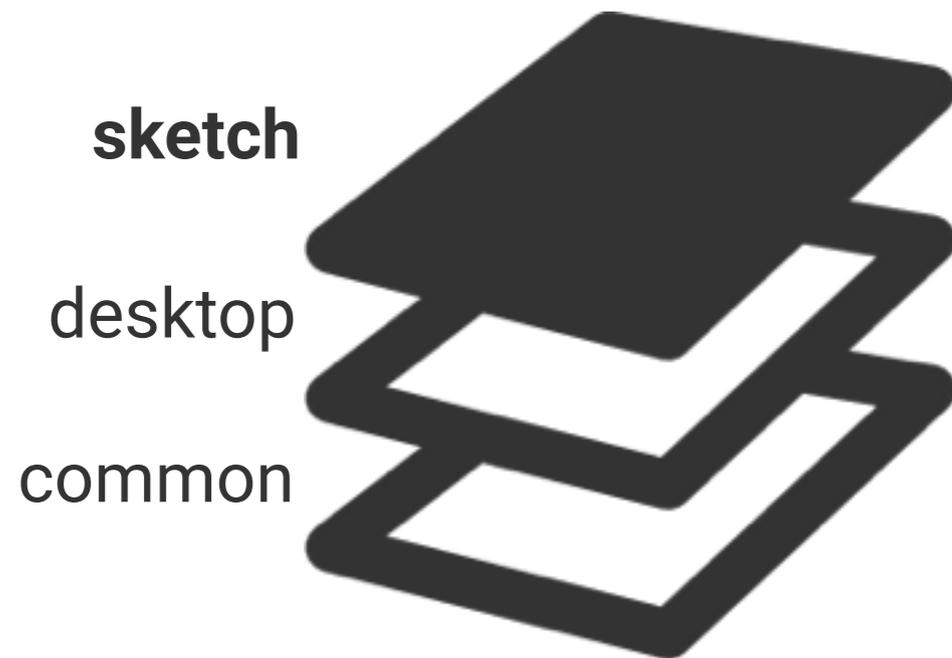
JSON

```
{  
  block : 'button',  
  content : 'Кнопка'  
}
```

SVG

```
<g class="button">  
  <rect class="button__instance"/>  
  <text class="button__text">  
    Кнопка  
  </text>  
</g>
```

Новая схема наложения уровней кода



CSS



Структура

- ▼ **button**
 - ▼ **__text**
 - # button__text.css
 - # button.css

CSS

common.blocks/button/button.css

```
.button {  
  font-size: 14px;  
  
  padding: 10px 20px;  
  
  border-color: #000;  
  border-width: 1px;  
  border-style: solid;  
  
  border-radius: 6px;  
  
  background: #fff;  
}
```

common.blocks/button/__text/button__text.css

```
.button__text {  
  font-size: 14px;  
  color: #000;  
}
```

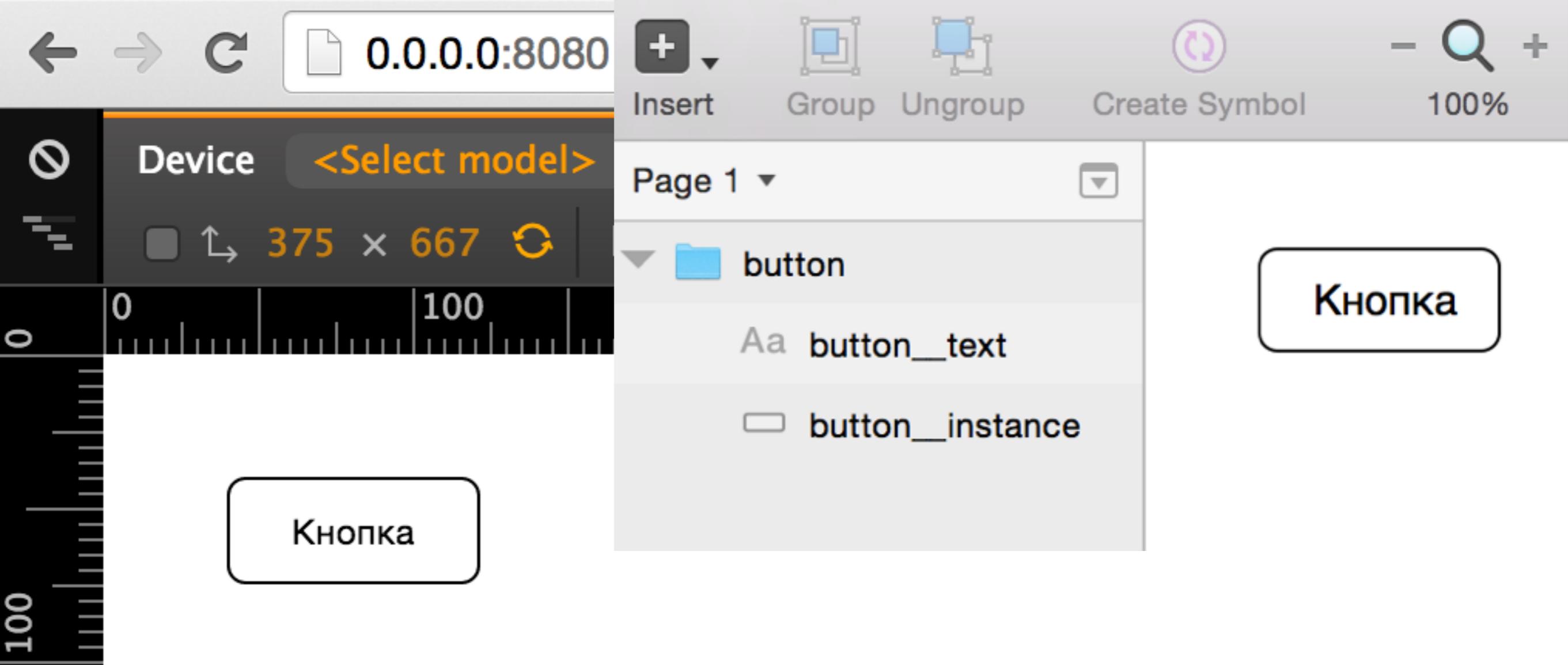
CSSTOATTRS

```
.button → <g class="button">  
.button__text → <rect class="button__instance"/>  
                <text class="button__text">  
                    Кнопка  
                </text>  
            </g>
```



PROFIT!!!

```
<g id="button">
  <rect id="button__instance" fill="none"
        rx="6"
        font-size="14"
        stroke="#000" stroke-width="1"
        width="80" height="34">
  </rect>
  <text id="button__text" fill="#000"
        font-size="14"
        x="18"
        y="22">
    Кнопка
  </text>
</g>
```



IT WORK!

bit.ly/svg-xmpl

Нет предела совершенству

ENB

bit.ly/enb-techs





P

Protein
Toolkit

the

P



the

P



save

the



save → **watcher**

the



save → watcher → **parse**

the



save → watcher → parse → **commit**

the



save → watcher → parse → commit → **push**

the



save → watcher → parse → commit → push → **PR**

the

P



the



PR

the



PR → **hook**

the



PR → hook → **watcher**

the



PR → hook → watcher → **pull**

the



PR → hook → watcher → pull → **rebuild**

the



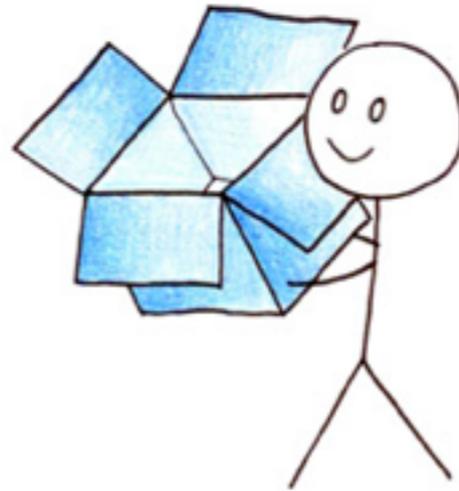
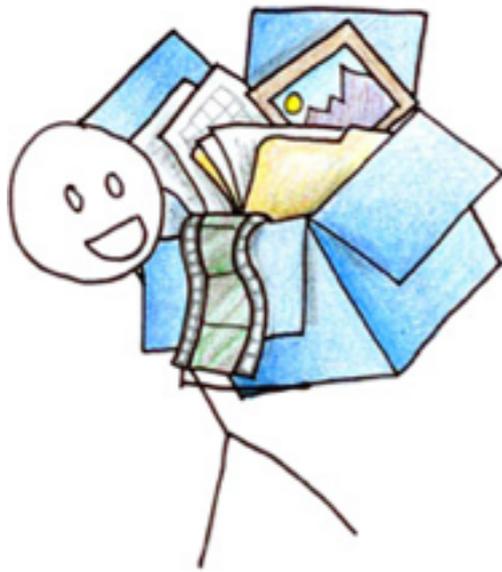
PR → hook → watcher → pull → rebuild → **reload**

the

P

Protein

Front end development workflow



the

P

Protein

the

Проблемы???

Проблемы

Проблемы

- разбор составных CSS свойств

```
.button {  
  font-size: 14px;  
→ padding: 10px 20px;  
  border-color: #000;  
→ border-width: 1px;  
  border-style: solid;  
→ border-radius: 6px;  
  background: #fff;  
}
```

Проблемы

- разбор составных CSS свойств
- размерность компонентов

sketch.blocks/button/button.css

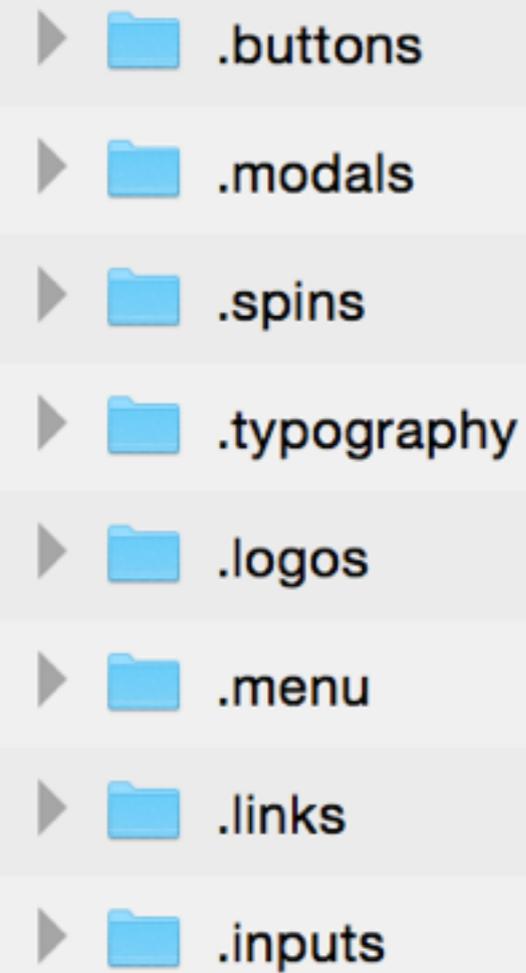
```
.button {  
  width: 80px;  
  height: 34px;  
}
```

Проблемы

- разбор составных CSS свойств
- размерность компонентов
- позиционирование

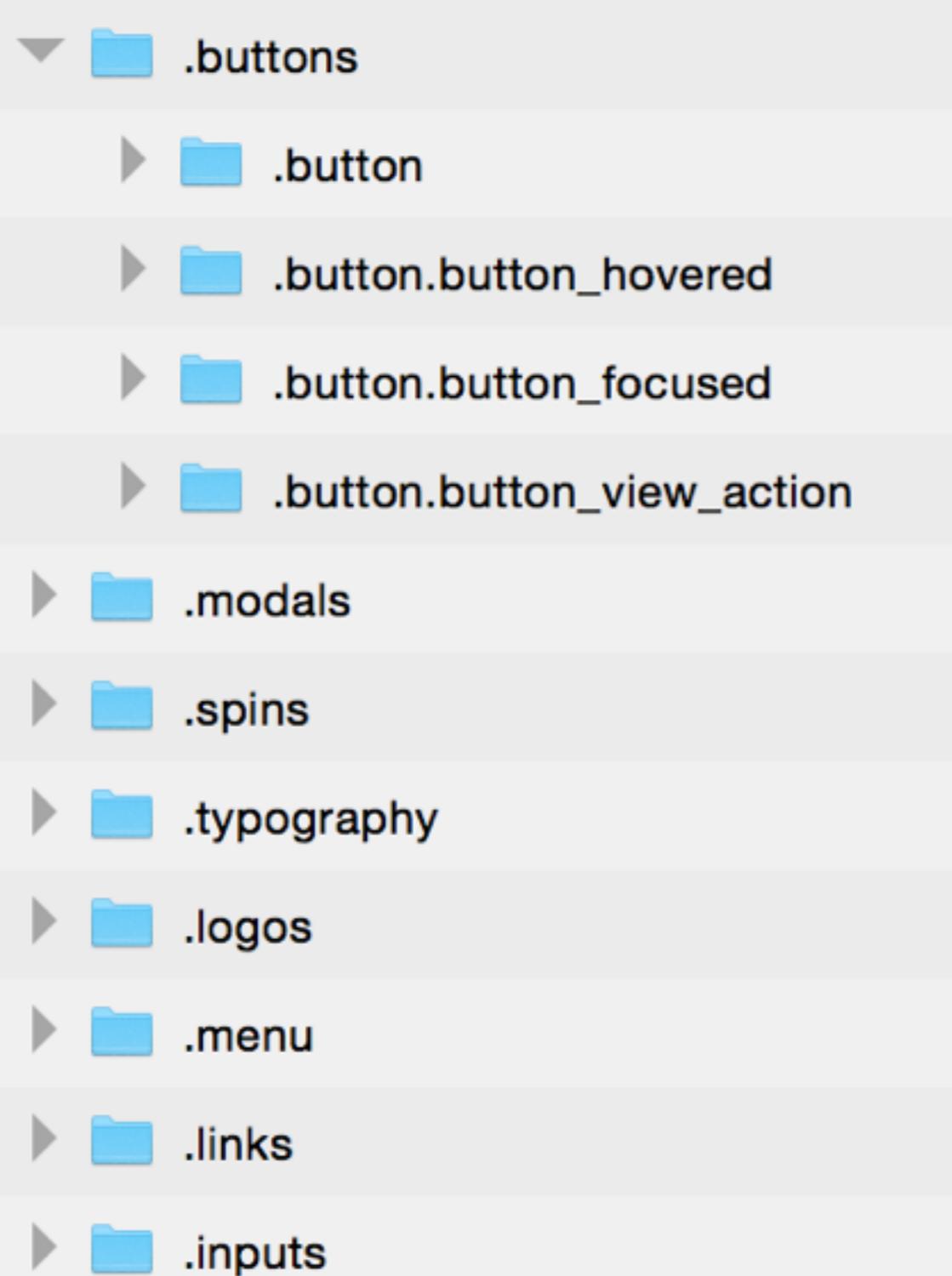
```
sketch.blocks/button/__text/button__text.css
```

```
.button__text {  
  x: 18px;  
  y: 22px;  
}
```



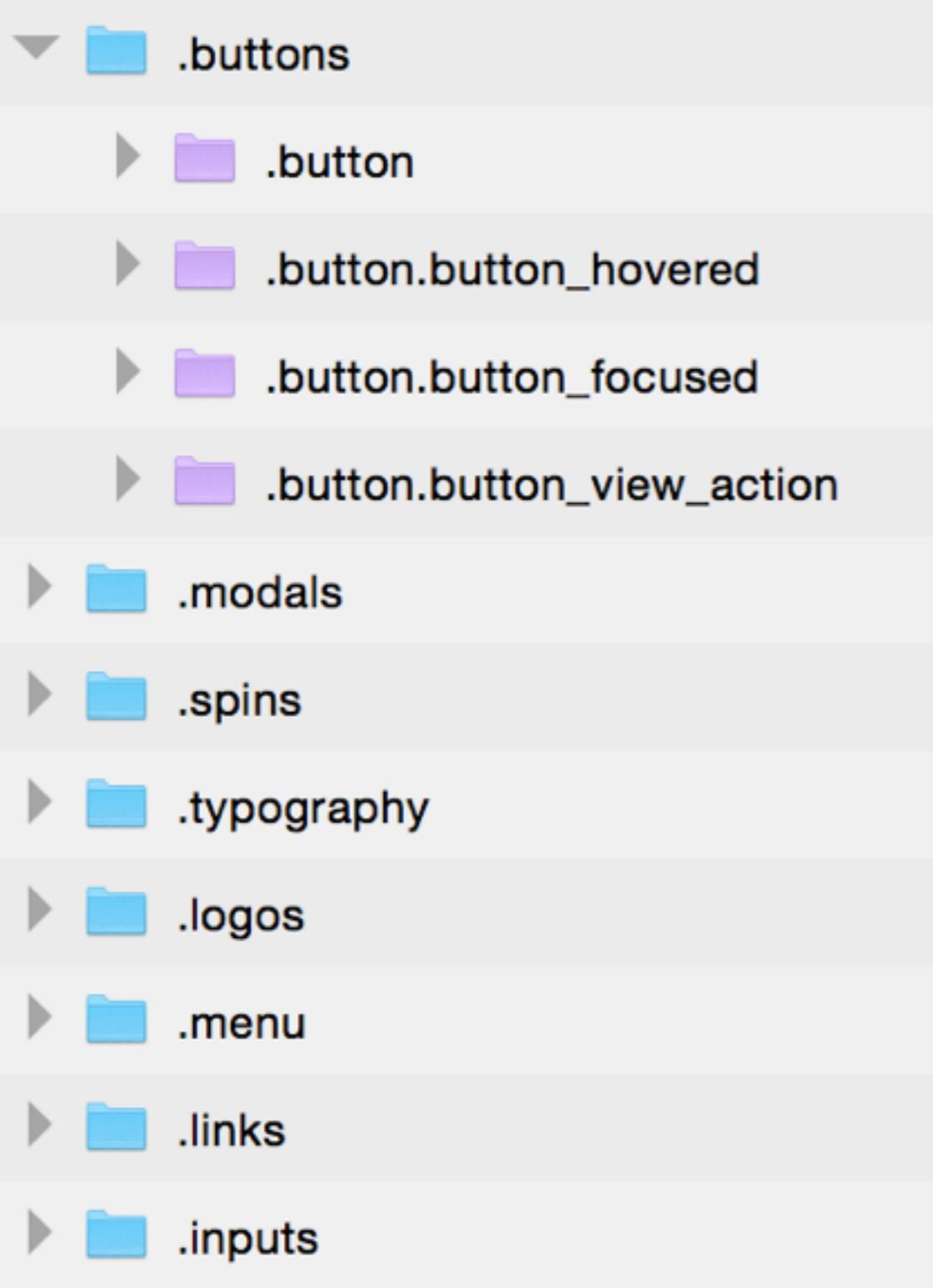
Проблемы

- разбор составных CSS свойств
- размерность компонентов
- позиционирование
- именованя



Проблемы

- разбор составных CSS свойств
- размерность компонентов
- позиционирование
- именованя
- разбор состояний



Проблемы

- разбор составных CSS свойств
- размерность компонентов
- позиционирование
- именованя
- разбор состояний
- программная специфика:
атрибуты, символы,
ограничения

Результат

- **одна** библиотека компонент
- **нет** кучи дизайнерских версий
- **всё** под контролем
- **все** в теме
- **один** язык общения
- **нет** дублирования работы



Спасибо!

theprotein.io

github.com/theprotein/svg-example

bem.info

bit.ly/fconf-comp-design

@awinogradov

