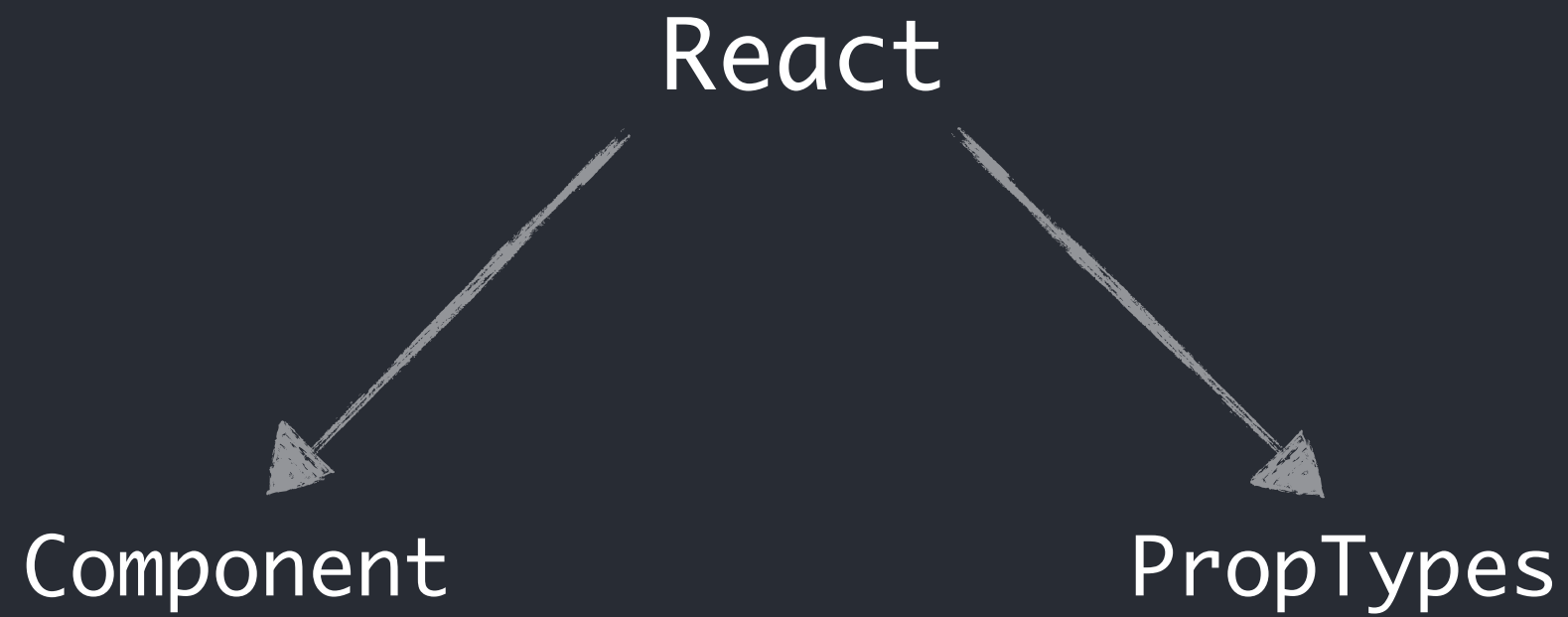


<Element />

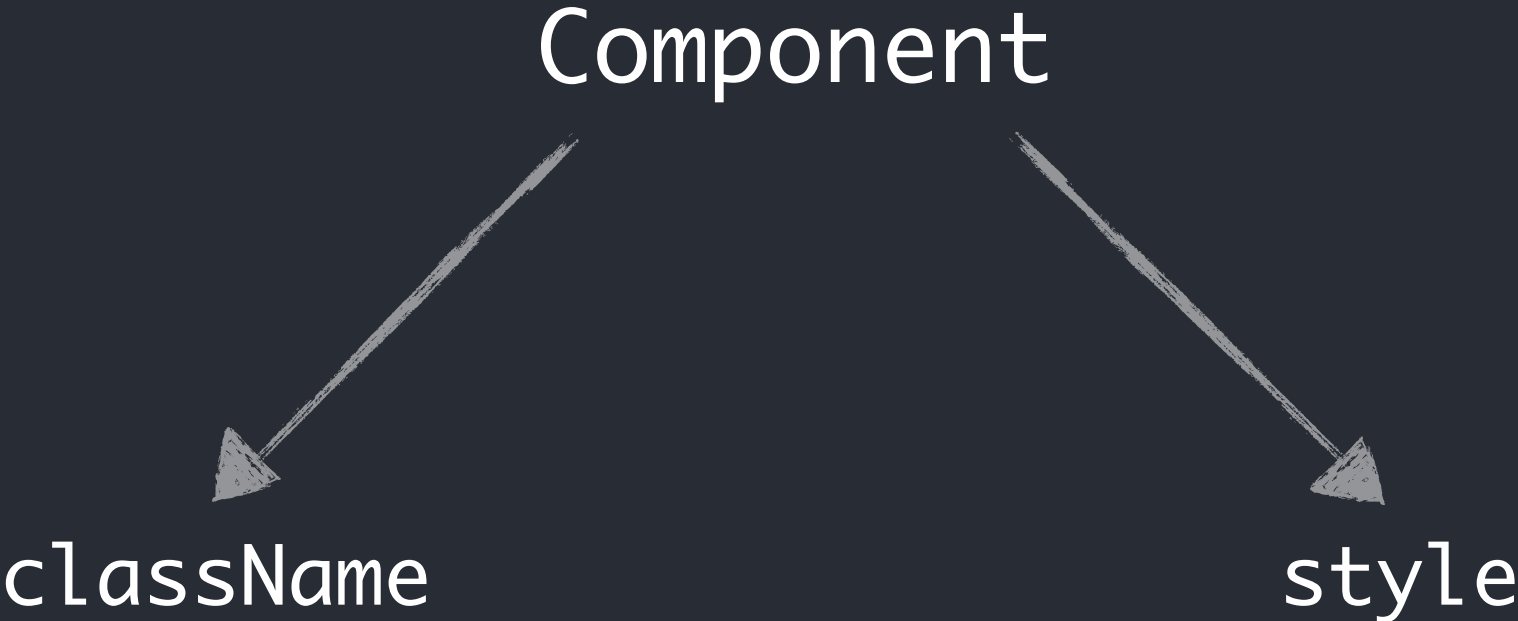
eleme/element-react

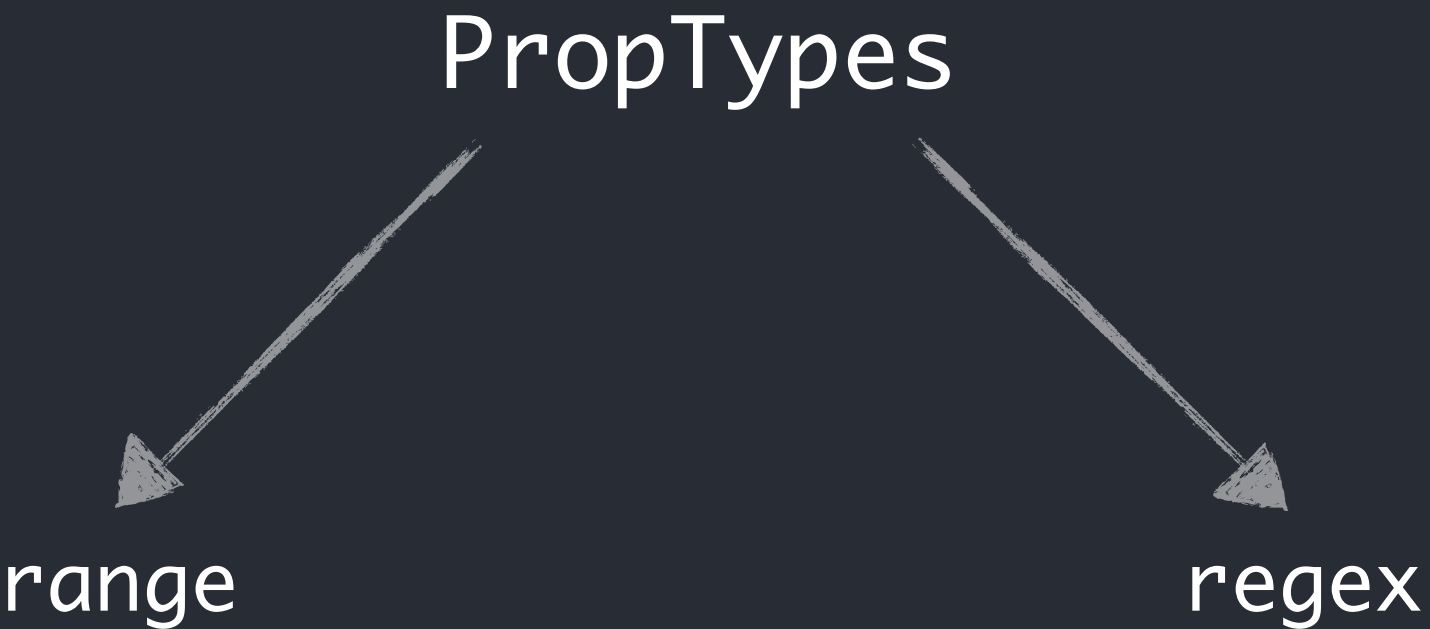
Element, 一套为开发者、设计师和产品经理准备的基于 Vue 2.0 的组件库, 提供了配套设计资源, 帮助你的网站快速成型。

element.eleme.io



```
class Button extends React.Component {  
  render() {  
    return <button>{this.props.children}</button>  
  }  
}  
  
Button.propTypes = {  
  type: React.PropTypes.string  
};
```





```
import Markdown from '../../libs/markdown';  
  
import './style.scss';  
  
export default class Button extends Markdown {  
  document(locale) {  
    return require('../../docs/${locale}/button.md');  
  }  
}
```

这段代码将Markdown文档渲染成HTML，并添加demo-button的class类

```
require(['../../src'], Element => {  
  const args = ['context', 'React'], argv = [this, React];  
  
  for (const key in Element) {  
    args.push(key);  
    argv.push(Element[key]);  
  }  
  
  args.push(this.component);  
  
  ReactDOM.unmountComponentAtNode(div);  
  ReactDOM.render(new Function(...args).apply(null, argv), div);  
});
```

Markdown的实现利用了一个简单的JVM， 将所有的组件都注入到文档中


```
import { transform } from 'babel-standalone';

this.component = transform(`
  class Demo extends React.Component {
    ${source[2]}
  }

  __rtn = (function() {
    return <Demo {...context.props} />
  })();
`, {
  presets: ['es2015', 'react']
}).code.replace('__rtn = ', 'return ');
```

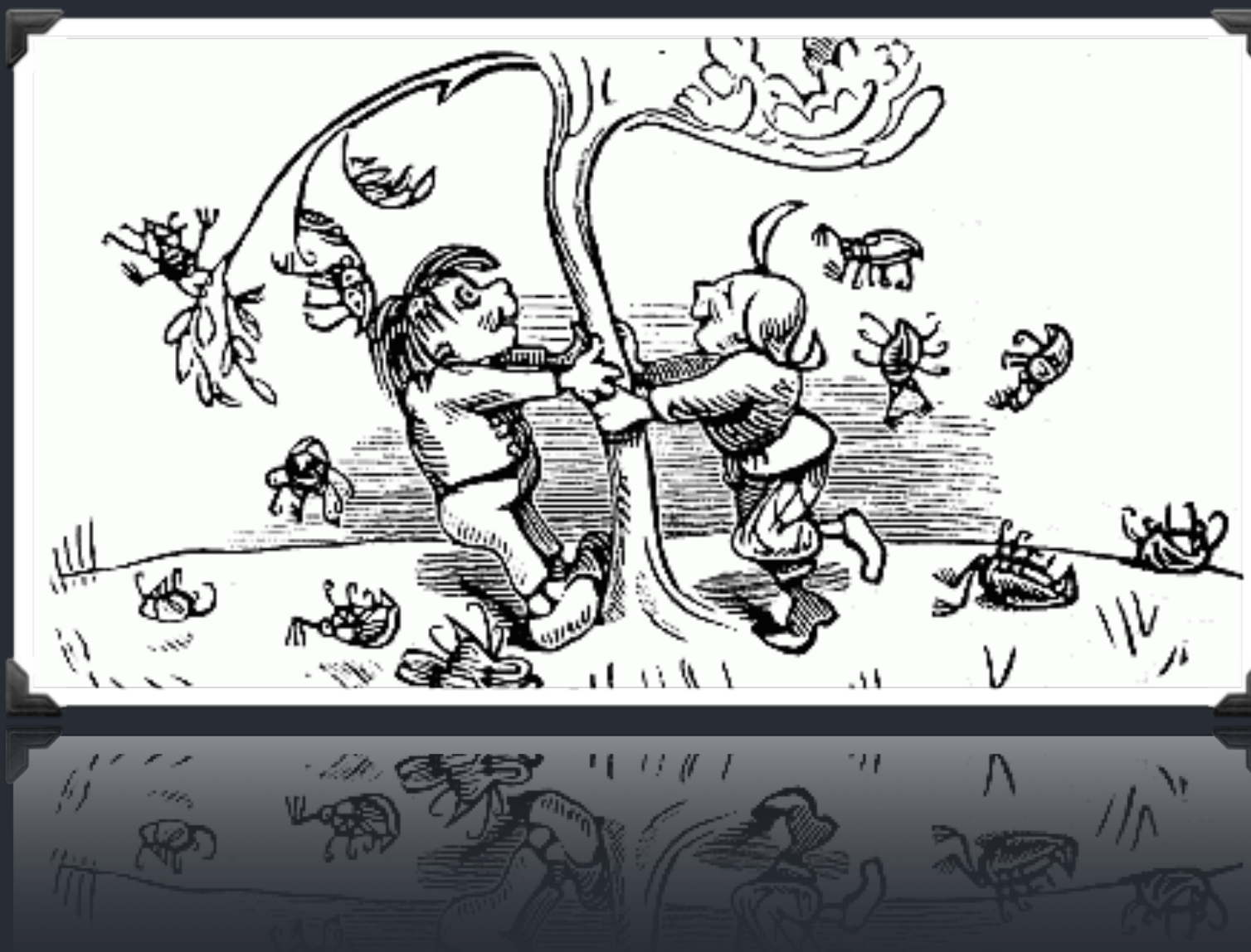
将Markdown文档中的React代码封装成可以执行的Component

```
new WebpackDevServer(webpack({
  devtool: 'eval',
  entry: [
    'webpack-dev-server/client?http://localhost:3000',
    'webpack/hot/only-dev-server',
    'react-hot-loader/patch',
    './index'
  ],
  plugins: [
    new webpack.HotModuleReplacementPlugin()
  ],
  ...
})
```

开发环境大家用到的应该都是react-hot-loader

```
entry: {
  site: path.join(basePath, 'site')
},
output: {
  path: path.resolve(basePath, 'dist/site'),
  chunkFilename: '[chunkhash:12].js',
  filename: '[chunkhash:12].js'
},
plugins: [
  new ExtractTextPlugin({ filename: '[chunkhash:12].css' }),
  new HtmlWebpackPlugin({ template: './index.html' })
].concat(process.env.TRAVIS_CI ? [] : [
  new webpack.DefinePlugin({
    'process.env.NODE_ENV': JSON.stringify('production')
  }),
  new webpack.optimize.UglifyJsPlugin({
    mangle: {
      keep_fnames: true
    },
    output: {
      comments: false
    }
  })
])
```

这是构建在线文档的webpack.config.js



Tree Shaking

```
// maths.js
```

```
export function square(x) {  
  return x * x;  
}
```

```
export function cube(x) {  
  return x * x * x;  
}
```

```
import { cube } from './maths.js';
```

```
console.log(cube(5)); // 125
```

```
function(e, t, n) {  
  "use strict";  
  
  function r(e) {  
    return e * e * e  
  }  
  t.a = r  
}
```

```
function(e, t, n) {  
  "use strict";  
  
  Object.defineProperty(t, "__esModule", {  
    value: !0  
  });  
  var r = n(0);  
  console.log(n.i(r.a)(5))  
}
```

Thanks

e1emeb0t