Public :

Assert:静态资源的配置目录

Svg：图标的查找地址

Index.html:访问目录

Src:

Canvas: 画板功能

Element-types:组件渲染

Arrange: 用于渲染组件四个角的

Box-element:暂时没用

Button-element:按钮组件

Container-elemnet:暂时没用

Coxcomb-element: 南丁格尔图组件

Form-element表单组件

FormItem-elemnet:表单项组件

Header-elemnet:暂时没用

Image-element:图像组件

Layout-element:暂时没用

List-element:列表组件

Table-element:表格组件

Text-element:文本组件

Index.js 组件export

Elements: 组件数据结构定义

Elemetnt.js 定义Element对象（扩展的js object对象），用于组件数据的序列化和反序列化

Form.js 继承Element对象,扩展表单数据结构的序列化和反序列化

Image.js继承Element对象,扩展图像数据结构的序列化和反序列化

slide.js继承Element对象,扩展根节点的序列化和反序列化

text.js继承Element对象,扩展文本的序列化和反序列化

index.js 定义组件的数据结构

HRuler:上游标

Property-ediotr:属性编辑器

Base-property：公有的属性编辑，如长宽高等

Button-propety:按钮的属性编辑器

Coxcomb-propety:南丁格尔树的编辑器

Event-property:事件的编辑器作废了

Event-templete:事件的编辑器

form-propety:表单的属性编辑器

image-propety:图像的属性编辑器

property-templete:根据属性类型定义的一些模板

silde-property: 根节点的属性编辑

table-propety: 表格的属性编辑器

text-property:文本属性编辑器

index.js：用于控制显示哪个属性编辑器

Tools: 工具栏

Util:工具类

TiletUTil.js 字符串操作的工具类

vRuler:左标尺

canvas-element.js:用于已绘制到画板上的组件拖拽和缩放大小

componentTree.js 用于显示组件树

componetTreeNode.js 用于显示组件树节点

constraints.js:这是计算定位啥的,很复杂

draggable.js 控制已绘制到画板上组件拖拽

index.js画板的主入口，部署是在这设置的

nodeinfo.js （忘了是干啥的了，貌似是因为拖拽功能对antd封装的树不支持，做了扩展）

potentialError.js 用于捕获组件异常的，不要因为一个组件的错误传播到整个react树

resizable.js 控制已绘制在画板上的组件缩放的

resize-node.js绘制画板上组件的四个缩放边

slide.js 绘制的组件的根，最好换个名字

Constants:常量信息

Element-list:组件选择界面

Element-item.css

Element-iitem.js

Httpservice：发送后端请求

Icon：SVG图标访问路口

Resource:资源管理配置的，暂时没用

Store:数据状态管理的

App.js React加载的主入口

Desginer.js 设计器入

Index.js js程序执行的住入口

Provider.js 用于传递数据状态