

The Design and Implementation of Responsive Web Front-End Framework

Wang Fei^{1,*}, Wang Sunhong²

¹College of Information Technology Beijing Normal University Zhuhai Campus, Zhuhai, China

²Guangzhou Konggu Technology Co., Ltd, Guangzhou, China

*wangfeizhuhai@aliyun.com

Keywords: Front-End Framework, Responsive Web, Componet

Abstract: Following the trend of Web front-end technology, a responsive framework of Web front-end has been designed and accomplished by using technology of media query, flow layout, responsive image, component modularization, data cache and animation engine. Since this framework provides rich components, it can help developer to create Web page by the means of free permutation and the problem of Web page adaptation can be resolved effectively. For the users who have special need, this framework provides strong extensibility so that each component can define various pattern parameters and combine better with real Web projects.

1. Introduction

Along with the developing of technique of Web front-end, Web page is becoming very complex [1]. More people access Internet through mobile terminal so that smart mobile devices occupy increasingly most market. Nowadays, Web page design need a well organized design method to make pages to be write once, run anywhere and provide an effective and excellent interactive experience [2]. There are several front-end frameworks to help developer to get this goal, such as Bootstrap [3] and AmazeUI. Based on HTML and CSS3, Bootstrap provides design templates, interface components, and selectable JavaScript components [4]. The biggest feature of Bootstrap framework is providing responsive grid system which can make special message blocks appear regularly on the page. However, since styles and patterns of the page designed by Bootstrap is much the same, the innovation of vision and interactive design can't be achieved while this framework be used by page developers. AmazeUI is an open source and cross-screen framework which is based on HTML5 and provides common component style [5]. AmazeUI is not popular in the Web front-end development, because it don't provide enough components. Therefore, following the trend of Web front-end technology, we investigate key technology of front-end responsive design, developing technology of front-end framework, and advantages of other same kind frameworks. Finally, a responsive framework be designed and implemented which can help developer to write elegant page, formulate feasible requirements for various Web site.

2. Requirement Analysis of the Framework

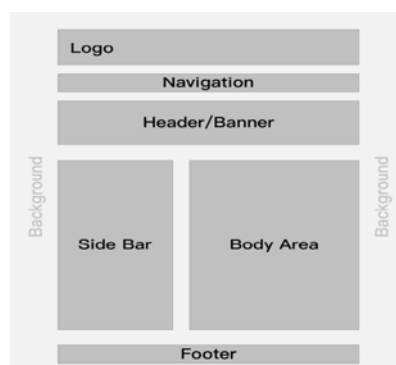


Figure 1 Example of page layout components

The framework provides a set of UI components which have properties of universality, consistency, and extensibility so that front-end codes can be reused for the uniform style purpose. These components include basic components, layout components, navigation components, and interactive components, etc. Basic components include header, paragraph, etc. Page layout components include flex-box, grid, column, margin, padding and divider, etc. as shown in Figure 1.

Page element components include list, table, form, and icons, etc. Interactive event components include pop-up windows, scroll-spy, smooth scroll, and tool tip, etc. which are carriers of program functions. Requirement description of these components in detail is shown in TABLE I. (requirement descriptions of other components are omitted).

Table 1 Requirement description of interactive event components

Component	Requirement Description
Pop-up windows	Pop-up window can be used as a floating layer to carry corresponding operations
Drag	Drag is a convenient way to move, copy, and delete page components
Toggle	Two status switching
Smooth scroll	Let users be quickly located in designated area
Scroll-spy	According the scroll actions to trigger related events
Notify	Globally display notice information

3. Implementation of the Framework

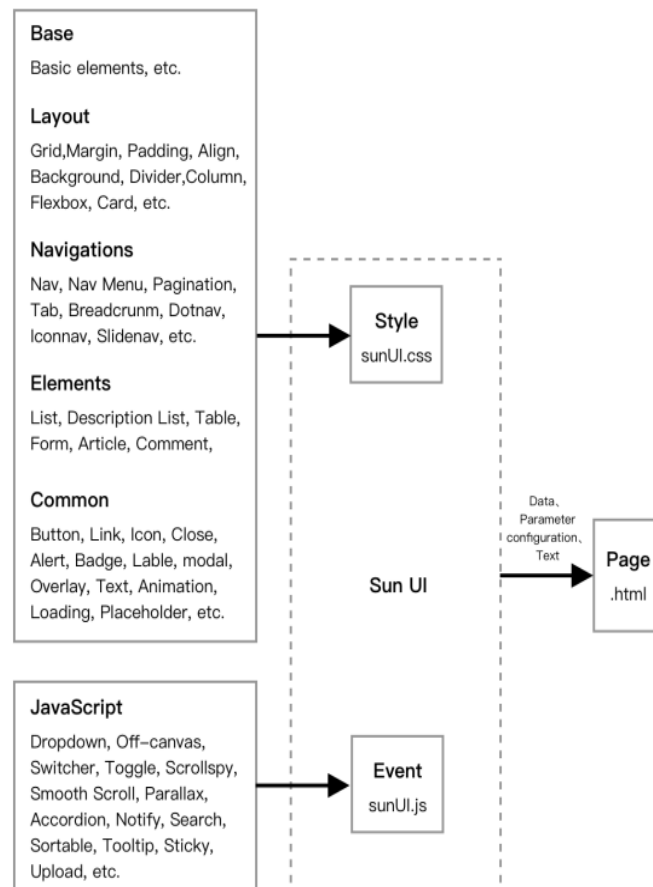


Figure 2 The structure of the framework

Based on the requirement investigation stated above, a scalable component framework has been implemented which is named as SunUI and is shown in Figure 2. This framework is mainly

consisted of two files. One is sunUI.css which is used to control styles and the other is sunUI.js which is used to control interactive events. Each file is collection of components. Through this framework a responsive web page can be generated by quoting components to accomplish page layout, parameter configuration, and text content.

In order to implement this framework, some main technique problems have been resolved, such as component modularization, data cache, animation engine, and drawing icon, etc. The technique details of component modularization and data cache are illustrated as fig.2.

3.1 Component Modularization

Modules can provide structure for large scale programs by partitioning code into different files and name space. Defining good module's interface can let modules be used and reused easily. With its flexible functions and objects, JavaScript defines good modularization system [6]. Function scope is used as inside space name of modules and objects is used to store a series of export values. At present, there are two modularization schemes which are defined well and widely used. One is CommonJS [7] and the other is AMD (Asynchronous Module Definition) [8-9]. In AMD , the “define” function, it accepts a module name array and a function. When loading finished, the “define” function is run and the argument which is module interface is passed to it at the same time. This framework adopts AMD to define program which need rely on other modules, as bellow:

```
(function (global, factory) {
  typeof exports === 'object' && typeof module !== 'undefined' ? module.exports
  =factory(require('jquery')) :
  typeof define==='function' &&define.amd ? define('sunUI', ['jquery'], factory) :
  (global.sunUI = factory(global.jQuery));
})(this, (function ($$1)
```

The “define” function loads dependency in bank-end so that obtaining module file and running page are at the same time. As soon as loading dependency is finished, “define” function call specified function and let these dependency modules as arguments be passed to the function. Through component modularization, components can be placed in any position of page therefore programs not need to move back and forth meanwhile avoid being organized in the structure which is not fit actual situation. The process of component modularization is mainly to declare component names and define a singleton object which provides access for public setting and action. Components simplify interaction by singleton pattern and act as a center warehouse of information and action. The inner functions of this singleton object include constants and values applied on page components which cooperate and provide functions for each other and allow users to initialize and manage these functions. The following code shows general structure and invocation method in component lifecycle:

```
//popup windows component
//declare JavaScript class
function Alert (sunUI){
  sunUI.component('alert',{
//create local array
    mixins:[Class,Toggable],
    args:'animation',
    props:{ animation:Boolean,
      close:String},
//declare options and defaults
    defaults:{ animation:true,
      close:'.sun-alert-close',
      duration:150,
      hideProps:{ opacity: 0}},
//define events
    events:[{name: 'click',
```

```

delegate:function delegate(){
  return this.close;},
handler: function handler(e){
  e.preventDefault();
  this.closeAlert();}},
//define methods
methods:{
closeAlert: function closeAlert(){
  var this$1 = this;
  this.toggleElement(this.$el).then(function(){
    return this$1.$destroy(true);});}});}

```

3.2 Data Cache

Developer usually need to save some state information for every component instance, such as option value, current usable state, and reference to other elements. It must be easy to access detail information of each target element. Data cache mechanism is the solution for this problem in this framework. The realization of data cache function has been completed by method in which custom data is attached to DOM element. This method is provided by HTML5 and is a new character of HTML5 for attribute of custom data. In this framework, framework's attribute configuration is completed by custom data. Meanwhile in order to use custom data easily and succinctly, data-prefix is hidden and sun- framework's prefix is used as identifier.

```

if(name.lastIndexOf('sun-',0) ===0 || name.lastIndexOf('data-sun-',0) === 0) {
name=camelize(name.replace('data-sun-',").replace('sun-',"));
if (sun[name]) {sun[name](node); } }

```

Hidden information or meta data related to specific element of page require to be stored. These data is used to set CSS style for DOM or set various values for applying script components in specific DOM elements. Every element can be set one or more attributes. In the data cache model, specified attribute value of element can be returned by using `getAttribute()` method in JavaScript and then corresponding manipulate of style or event can be proceeded.

```

var options=el.getAttribute(this.$name) || el.getAttribute ("data-"+(this.$name))) ,key,prop;
parse the captured value and map to the regulation of corresponding style or event:

```

```

if(options[0]==='{'){
  try{
    options=JSON.parse(options);
  }catch(e){
    console.warn("Invalid JSON.");
    options={};}
  }else if(args.length&& !~options.indexOf(':')){
    options=((obj={},obj[args[0]]=options,obj));
    var obj; }else{
    var tmp ={};
    options.split(';').forEach(function(option){
      var ref=option.split(/:(.+)/);
      var key=ref[0];
      var value=ref[1];
      if (key && value){
        tmp[key.trim()]=value.trim();});
    options=tmp; }
    for (key in options||{}){
      prop=camelize(key);
      if (props[prop] !==undefined){
        data[prop]=coerce(props[prop],options[key],el); } }

```

4. The Implemented Components by This Framework

The implemented components by this framework include basic component, layout component, page element component, and interactive (JavaScript) event component. Implementation details of the design are illustrated only for grid and column of layout component and mouse drag event of interactive event component because of limited thesis length.

4.1 Implementation Details of Grid and column

Grid and column are a skeleton of responsive design. Grid and column can be created by using classes of "sun-grid" and "sun-column". When grid and column occupy a fixed percentage size of thire parent, they can extend and shrinky automatically to make them be a ideal component for flow layout, as shown in Figure 3.

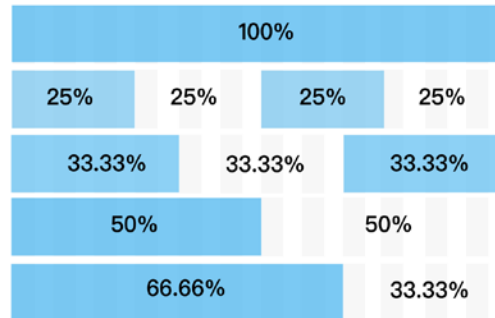


Figure 3 Grid and column sketch

The corresponding code is shown bellow:

```
//selector, set "sun-child-width" style regulation
[class*='sun-child-width']>{*{
  box-sizing:border-box;width:100%;}
.sun-child-width-1-2>{*{width:50%;}
.sun-child-width-1-3>{*{width:calc(100%*1/3.001);}
.sun-child-width-1-4>{*{width:25%;}
.sun-child-width-1-5>{*{width:20%;}
.sun-child-width-1-6>{*{width:calc(100%*1/6.001);}
.sun-child-width-auto>{*{width: auto;}
```

According media query regulation, different styles are setted to different resolution screen. This method is called breakpoint, as shown in Figure 4.

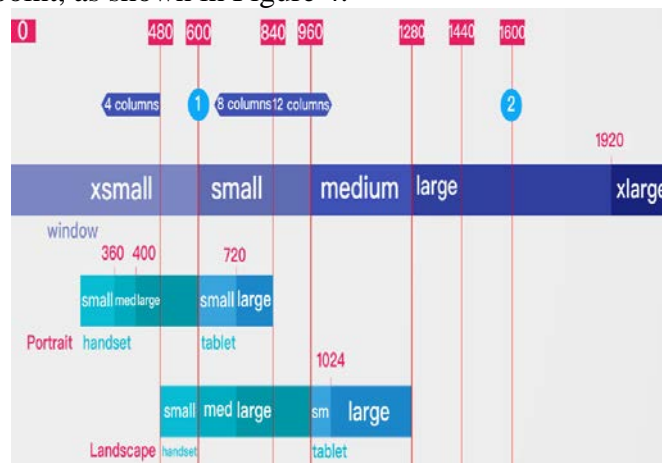


Figure 4 Breakpoints of grid

In This framework, there are four breakpoints:640px, 900px, 1200px, 1600px. There are different styles corresponding to different resolution range:

```
/* normal style */
tag,class,id,or name{/* CSS regulation */}
```

```

/* Phone landscape and bigger */
@media(min-width:640px){/* CSS regulation */}
/* Tablet landscape and bigger */
@media(min-width:960px){/* CSS regulation */}
/* Desktop and bigger */
@media(min-width:1200px){/* CSS regulation */}
/* Large screen and bigger */
@media(min-width:1600px){/* CSS regulation */}

```

4.2 Implementation Details of Drag Component

Drag component is packaged on several mouse events of target element. Since low-level events of drag component are transformed to high-level events directly related to drag operation, drag component only need to realize drag actions which they required instead of caring about low-level events such as how to detect in the initial point, when to start a drag and how to track drag progress. Sequence diagram of drag component is shown in Figure 5.

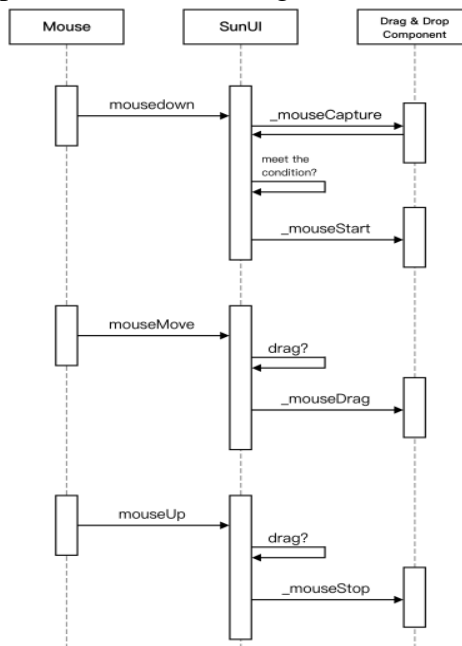


Figure 5 Sequence chart of drag component

Drag component create following drag events for user to operate:

- dragged object start: mouse starts to move.
- dragged object leave initial point: dragged page element leaves initial point.
- dragged object return initial point: dragged page element return to initial point.
- dragged object arrive target position: dragged page element enter target position.
- dragged object leaves target position: dragged page element leaves target position.
- dragged object is accepted: dragged page element can be placed on target position.
- dragged object is rejected: dragged page element can't be placed on the target position.

Drag operation create a series of events to realize re-permutation of page elements and to change object dependency.

5. Conclusion

Depending on the basis of investigation of Web front-end trend, key technique of responsive design, developing technique of front-end framework, a responsive front-end framework is implemented. This framework include rich components to help developer create Web pages by the means of free permutation and the problem of Web page adaptation can be resolved effectively. For the users who have special need, this framework provides strong extensibility so that each

component can define various pattern parameters and combine better with real Web projects. It is troublesome to use Bootstrap framework, because a lot of re-write CSS is needed to improve no-Bootstrap styles and result in helplessness of using Bootstrap. In the future, there are many work will be done continuously, such as component optimizing and function extension, diverse themes, personalized parameter configuration, and visual building tools for Web pages, etc. Responsive Web page design is not only a design philosophy, it also bring about brand new development model. How to plan page content for any size screen and how to deploy development process are becoming new tasks.

References

- [1] Zhicheng, Dai., Jingcao, Cheng.: 'Research of optimization for Web front-end performance based on virtual DOM', Computer Applications and Software, 2017, 34, (12), Dec. pp. 21-25
- [2] Fei, Chen.: 'The technology of Web front-end and analysis of optimizing direction', New Media Investigation, 2015, 1, (3), pp. 39-40
- [3] 'Bootstrap global CSS style', <http://v3.bootcss.com/css/>, accessed 17 January 2013
- [4] Frain, B.: 'Responsive Web design with HTML5 and CSS3' (Packt Publishing Limited, 2012, 1st edn.), pp.1-17
- [5] Cheng, Wang. Shaoyuan, Li., Lixiao, Zheng. 'Optimisation scheme for Web front-end performance and its practice', Computer Applications and Software, 2014, 31, (12), pp.89-95
- [6] Obcena, M.J.: 'JavaScript and MooTools', (Apress, 2010), pp.36-48
- [7] Xiaoyue, Hu. 'The design and Implementation of a Web front-end UI framework', Master's thesis, Huazhong University of Science and Technology, 2016, pp.5-6
- [8] Nanpin, Chen: 'The Design and Implementation of Asynchronous Loading AMD Engine', Software Guide, 2014, 13, (11), pp.105-107
- [9] 'Asynchronous module definition', https://en.wikipedia.org/wiki/Asynchronous_module_definition, accessed 2 February 2013