


```

config4frame = {
  resizable: false, //
  //
  north: {
    height: 80 //(60px)
    visible: true //
  },
  //
  south: {
    visible: false //
  },
  //
  west: {
    width: 220 //
  },
  //
  east: {}
}

```

```

config4MenuTree = {
  region: 'west', //
  onBeforeInit: null, //[element]
  onAfterInit: null, //[element]
  onDataFilter: null, //[node, childNodes]
  onBeforeNodeClick: null, //[node, $node, $li]
  onNodeClick: null, //[node, $node, $li]
  onAfterNodeClick: null, //[node, $node, $li]
  onBeforeNodeCreate: null, //[node, $node, $li]
  onNodeCreate: null, //[node, $node, $li]
  onAfterNodeCreate: null, //[node, $node, $li]
  onBeforeNodeExpand: null, //[node, $node, $li]
  onNodeExpand: null, //[node, $node, $li]
  onAfterNodeExpand: null, //[node, $node, $li]
  onBeforeNodeCollapse: null, //[node, $node, $li]
  onNodeCollapse: null, //[node, $node, $li]
  onAfterNodeCollapse: null, //[node, $node, $li]
  onBeforeDisplayNodes: null, //[node, $node, $li]
  onDisplayNodes: null, //[node, $node, $li]
  onAfterDisplayNodes: null //[node, $node, $li]
}

```

1. node -

```

node = {
  text: '', //entry
  id: '213', //entry id
  isexpand: false, //
  hasChildren: true, //
  ChildNodes: [], //
  isModule: false, //
  level: 1 //0
}

```

2. \$node - DOM

3. \$li - DOM

tab

```
config4tabPane = {
  region: 'east', //
  style: 'alpha', //alphabravo
  tabWidth: 150, //tab
  isCollapsible: true, //
  hasHomepageBtn: false, //
  onCreateTab: null, //Tab():[$tab, entry]
  onSelectTab: null, //Tab():[$tab, $content, entry]
  onCloseTab: null, //Tab():[$tab, $content, entry]
  afterLoadTab: null, //Tab():[$tab, $content, entry]
}
```

```
FS.tabPane.addItem(item)
: tabtabtab
: item {JSON}
  item = {
    text: 'newTab', //, tab
    id: '213', //entry id
    src: 'http://www.finereport.com', //iframe
    onCreate: null, //Tab:[$tab, entry]
    onSelect: null, //Tab:[$tab, $content, entry]
    onClose: null, //Tab:[$tab, $content, entry]
    afterLoad: null, //Tab:[$tab, $content, entry]
  }
```

Navigation

```
config4navigation = {
  onBeforeInit: null, //
  onAfterInit: null, //
  naviComponents: null //, push, renderonClick;
}

naviComponents = [ //push
  navi : {...}, //
  search: {...}, //
  reg: {...} //
]
// naviComponents.item
{
  renderEl: JQuery
  render:Function
  onClick:Function
  visible:Boolean
}
naviComponentsclassid
```

Gallery

```
config4Gallery: {
  region: null //
}
```

png179499938

```
FS.Plugin.CustomIcon.push({
  //
  getMenuNodeCustomIcon: function (icon) {
    return null;
  },
  //
  addCustomIconItem : function(configSetting){
  },
  //
  removePreCustomIconSelected : function(folders) {
  },
  //
  addItem2Folders : function(eachFolder, $folders) {
  },
  //
  initSelectedCustomIcon : function(selectedFolder) {
  },
  //ID
  getCustomIconAttachID : function (folder) {
    return null;
  }
});
```

jscss

theme.jsFS.Theme

```
(function ($) {
  FS.THEME = $.extend(true, FS.THEME, {
    /***/
  });
})(jQuery);
```

Classic

```

{
  config4frame: {
    resizable: true,
    //
    north: {
      visible: true //
    },
    //
    south: {
      visible: false //
    },
    //
    west: {
      width: 230 //
    },
    //
    east: {}
  },
  config4Gallery: {
    region: null
  },
  config4MenuTree: {
    region: 'west'
  },
  config4tabPane: {
    region: 'east',
    style: 'bravo',
    isCollapsible: false,
    hasHomepageBtn: true
  },
  config4navigation: {
    onBeforeInit: null,
    onAfterInit: null
  }
}

```

Modern

```

{
  config4frame: {
    resizable: false,
    south: {
      visible: true
    }
  },
  config4Gallery: {
    region: 'east'
  },
  config4MenuTree: {
    onNodeExpand: function (node, $node, $parent) {
      return node.level !== 0;
    },
    onNodeClick: function (node, $node, $parent) {
      $('#fs-frame-wrapper') .empty();
      return false;
    }
  },
  config4tabPane: {
    style: 'alpha',
    region: 'south',
    isCollapsible: true,
    hasHomepageBtn: false
  }
}

```

Demo:

style.css

```
.node-navi{
  position: relative;
  float: right;
  right: 30px;
  list-style: none;
  height: 60px;
  top: 0;
}

.node-navi li{
  position: relative;
  float: left;
  left: 0;
  display: block;
  height: 60px;
  line-height: 60px;
  color: #fff;
  font-size: 14px;
  padding: 0 15px;
  cursor: pointer;
}

.node-navi li:hover{
  color: #6fd3ff;
}

.node-pane{
  position: absolute;
  top: 60px;
  left: 0;
  -webkit-border-radius: 0 0 3px 3px;
  -moz-border-radius: 0 0 3px 3px;
  z-index: 10000;
}

.node-pane-inner{
  position: relative;
  height: 200px;
  overflow-x: hidden;
  overflow-y: auto;
  background: rgb(41, 90, 148);
  background: rgba(41, 90, 148, 0.85);
  *width:200px;
}

.node-select{
  background: #295a94;
}

.node-pane a, .node-title{
  position: relative;
  white-space: nowrap;
  text-overflow: ellipsis;
  overflow: hidden;
  display: block;
  min-width: 160px;
  _width:180px;
  height: 20px;
  line-height: 20px;
  font-size: 14px;
  color: white;
  padding: 3px 5px 3px 12px;
  cursor: pointer;
}

.node-pane a:hover{
  background: rgb(41, 90, 148);
}
```

```

}
.node-title{
padding: 5px 5px 5px 4px;
cursor: default;
color: #6fd3ff;
font-weight: bold;
}

```

theme.js

```

(function ($) {
    FS.THEME = $.extend(true, FS.THEME, {
        config4navigation: {
            onAfterInit: function () {
                var self = this;
                $('#fs-frame-search').remove();
                var $reg = $('#fs-frame-reg');
                if ($reg.length > 0) {
                    $reg.remove();
                }
            }
        }
    });
    $.ajax({
        url: FR.servletURL + "?op=fs_main&cmd=module_getrootreports",
        type: 'POST',
        data: {id: -1},
        success: function (res, status) {
            var nodes = $.parseJSON(res);
            $.ajax({
                url: FR.servletURL + "?op=fs_main&cmd=getmoduleitems",
                type: 'POST',
                async: false,
                data: {id: 1},
                success: function(res){
                    nodes.push($.parseJSON(res));
                }
            });
            var $ul = $('<ul class="node-navi"/>').appendTo($('#fs-frame-header'));
            $.each(nodes, function (index, root) {
                var $node = $('<li class="node-navi-li"/>').appendTo($ul);
                $('<div/>').text(root.text)
                    .appendTo($node)
                    .click(function () {
                        if($node.hasClass('node-select')){
                            return;
                        }
                        $ul.find('.node-select').removeClass('node-select');
                        $node.addClass('node-select');
                        var $dropdown = $(this).data('DATA');
                        if (!$dropdown) {
                            $dropdown = $('<div class="node-pane"/>').appendTo($node);
                            $(this).data('DATA', $dropdown);
                            var $pane = $('<div class="node-pane-inner"/>')
                                .css({
                                    'max-height': document.body.clientHeight - 90
                                }).appendTo($dropdown);
                            if (root.hasChildren && root.ChildNodes) {
                                var $other = $('<div class="node-wrapper"/>').appendTo($pane);
                                $.each(root.ChildNodes, function (index, child) {
                                    if (child.hasChildren) {
                                        var $w = $('<div class="node-wrapper"/>').appendTo($pane);
                                        $('<div class="node-title"/>').text(child.text).appendTo($w);
                                        var childs = [];
                                        _collectAllChildNodes(child, childs);
                                        $.each(childs, function (i, n) {
                                            _createItem(n, $dropdown, $node).appendTo($w);
                                        });
                                    } else {
                                        _createItem(child, $dropdown, $node).appendTo($other);
                                    }
                                });
                            }
                        }
                    });
            });
        }
    });
}

```

```

        } else {
            return;
        }
    }
    $dropdown.fadeIn('fast');
    $(document).bind('mouseover.nodepane', function (e) {
        var $t = $(e.target);
        if ($t.closest('.node-pane').length <= 0) {
            $node.removeClass('node-select');
            $dropdown.fadeOut('fast');
            $(document).unbind('mouseover.nodepane');
        }
    });
});
});
});
});
},
config4frame: {
    west: {
        width: 0
    }
}
});
var _createItem = function (node, $pane, $node) {
    return $('<a href="#">').text(node.text)
        .click(function () {
            FS.tabPane.addItem(node);
            $node.removeClass('node-select');
            $pane.hide();
            $(document).unbind('mousedown.nodepane');
        });
};
var _collectAllChildNodes = function (node, childs) {
    var self = this;
    if (!node.ChildNodes) {
        return;
    }
    $.each(node.ChildNodes, function (index, child) {
        if (child.hasChildren) {
            _collectAllChildNodes(child, childs);
        } else {
            childs.push(child);
        }
    });
};
})(jQuery);

```



theme.jsFS.Plugin.CustomIcontheme.js

```
(function($){
    //
    FS.THEME = $.extend(true, FS.THEME, {
        /****/
    });

    //
    FS.Plugin.CustomIcon.push({
        /**/
        getMenuNodeCustomIcon: function (icon) {
            return null;
        },
        /**/
        addCustomIconItem : function(configSetting){
        },
        /**/
        removePreCustomIconSelected : function(folders) {
        },
        /**/
        addItem2Folders : function(eachFolder, $folders) {
        },
        /**/
        initSelectedCustomIcon : function(selectedFolder) {
        },
        /**ID*/
        getCustomIconAttachID : function (folder) {
            return null;
        }
    });
})(jQuery);
```

demo

theme.js

```
demo

(function ($) {

    FS.Plugin.LookAndFeelSettings.push({
        item: function(ctx) {
            return {
```

```

        title: FR.i18nText(""),
        content: $("<div />")
    };
},
action: function (tabPane) {
}
});

FS.THEME = $.extend(true, FS.THEME, {
    /***/
});

FS.Plugin.CustomIcon.push({
    getMenuNodeCustomIcon: function (icon) {
        /*op,op*/
        /*cmd=read_img,png*/
        var config = {
            url: FR.servletURL + '?op=custom_icon&cmd=read_img',
            data: {
                id : icon
            }
        };
        var attach = FS.Sync.ajax(config);
        var $icon = $('<span/>').addClass("fs-menu-icon");
        /*cmd=getattach,pngimgalt*/
        var url = FR.servletURL + ('?op=custom_icon&cmd=get_attach&id=' + attach.attachid +
'&isAdjust=false');
        $('<img>').attr('src',url).attr('alt', attach.attachfile).addClass('fs-menu-node-icon').appendTo
($icon);
        return $icon;
    },
    removePreCustomIconSelected : function(folders) {
        //
        var curFolder = $('<span/>').selected .folder-icon', folders);
        $('<span/>').selected .fs-folder-customIcon-selected', folders).remove();
        curFolder.removeAttr('style');
    },
    addCustomIconItem : function(configSetting){
        var self = configSetting;

        var config = {
            url: FR.servletURL + '?op=custom_icon&cmd=read_img',
            data: {
                id : ''
            }
        };
        var completeFn = function (res, status) {
            var result = FR.jsonDecode(res.responseText);
            $.each(result, function(index, attach){
                var $icon = $('<div class="fs-icon-item"/>').hover(function () {
                    $(this).addClass('icon-hover');
                }, function () {
                    if (!$<div class="fs-icon-item"/>').hasClass('selected')) {
                        $(this).removeClass('icon-hover');
                    }
                }).click(function () {
                    self.selectIconItem($<div class="fs-icon-item"/>');
                });
            });
            var url = FR.servletURL + ('?op=custom_icon&cmd=get_attach&id=' + attach.attachid +
'&isAdjust=false');
            $('<img>').attr('src', url).attr('alt', attach.attachfile).addClass('fs-customIcon-list-
item').appendTo($icon);
            $icon.data('ENTRY', {text: attach,onSelect: function ($icon) {
                var curFolder = $('<span/>').selected .folder-icon', self.$folders);
                //
                $('<span/>').selected .fs-folder-customIcon-selected', self.$folders).
remove();
                curFolder.removeAttr('style');

                var url = $('<span/>').selected .folder-icon', $icon).attr('src');
                var altText = $('<span/>').selected .fs-folder-customIcon-selected', $icon).attr('alt');
            }
        });
    }
});

```

```

                var img = $('<img>').attr('src',url).attr('alt', altText).addClass('fs-folder-
customIcon-selected');
                curFolder.attr('style', 'padding-left:0px').html("");
                img.insertBefore($('<div class="fs-icon-folder-item.selected folder-icon">'));
            });
            $icon.appendTo(self.$icons);
        });
    };
    FS.Async.ajax(config, completeFn);
},
addItem2FolderNodes : function(eachFolder, $folders) {
    var config = {
        url: FR.servletURL + '?op=custom_icon&cmd=read_img',
        data: {
            id : eachFolder.value
        }
    };
    var attach = FS.Sync.ajax(config);
    if(attach.attachid && attach.attachfile) {
        var url = FR.servletURL + ('?op=custom_icon&cmd=get_attach&id=' + attach.attachid +
'&isAdjust=false');
        $('<img>').attr('src', url).attr('alt', attach.attachfile).addClass('fs-folder-customIcon-
selected').appendTo($folders);
        $('<i class="folder-icon">').attr('style', 'padding-left:0px').html("").appendTo($folders);
    }
},
initSelectedCustomIcon : function(selectedFolder) {
    var altText = $('<div class="fs-folder-customIcon-selected">').attr('alt');
    $.each(FSCS.View.$icons.children(), function(i, folderIcon){
        if (($('<div class="fs-customIcon-list-item">', folderIcon).attr('alt'))) {
            if (altText === $('<div class="fs-customIcon-list-item">', folderIcon).attr('alt')) {
                FSCS.View.selectIconItem(FSCS.View.$icons.children().eq(i));
            }
        }
    })
},
getCustomIconAttachID : function (folder) {
    //imgalt
    return $('<div class="fs-folder-customIcon-selected">', $(folder)).attr('alt');
}
});
})(jQuery);

```

style.css

demo

```

.fs-menu-node-icon {
    width: 25px;
    height: 16px;
}

.fs-folder-customIcon-selected {
    width:25px;
    height:16px;
    padding-left:10px;
}

.fs-customIcon-list-item {
    width:80px;
    height:80px
}

```

[demo](#)


```

.fui-bsb{
  background-color: @color;
}
.fui-bsc{
  color: @color;
}
.fui-bsd{
  border-color: @color;
}

```

```

.fui-fhc{
  color: @color;
}
.fui-fht{
  text-shadow: 0 0 3px @color;
}

```

```

.fui-seb{
  background-color: @color;
}

```

```

.fui-fbc{
  color: @color;
}
.fui-fbt{
  text-shadow: 0 0 1px @color;
}

```

FR.servletURL + "?op=fs_main&cmd=getmoduleitems"	{id: -1}

FR.servletURL + "?op=fs_main&cmd=module_getrootreports"	{id: -1}

BI

		callback
FR.servletURL + "?op=fr_bi&cmd=get_folder_report_list"	{}	[[{lastModify:1466402196217,pld:"-1","id":"1",text:"",value:"348602efbb80f686"}]]

BI

		callback
FR.servletURL + "?op=fr_bi&cmd=add_report"	{reportName:", reportLocation: 'id', realTime: false}	{"reportId":23}

BI

		callback
FR.servletURL + "?op=fr_bi&cmd=init_dezi_pane&reportId=23&edit=_bi_edit_"		

BI

		callback
FR.servletURL + "?op=fr_bi_configure&cmd=init_configure_pane"		

```
FS.loadModule(render, moduleName)

render dom
  moduleName
FS.loadModule($(' <div/>').appendTo('body'), 'report');
```

	report
	user
	privilege
	schedule
	sysmgr
	lookandfeel
	register
	monitor
	mobile

FS

```
FS.signOut()
```