Class MMoverlay inherit google.maps.OverlayView

property: map div\_

MMoverlay.prototype.onAdd = function() {} // overwritten this function

//mainly responsible for Exterior rendering

MMoverlay.prototype.draw = function() {} // init the position and set it as hidden

//overwritten function

MMoverlay.prototype.draw2 = function( latLng ) {...}

// In mapinteraction.js, when map is default mode, right click will call this function.

// This method is responsible for rendering the box at corresponding position.

//please pay attention, at this time, the box is still in hidden, In mapinteraction.js, after draw2(latLng) method, toggleOn() method will be called, that method will set box as visible

MMoverlay.prototype.onRemove = function() {}

// This is completely remove div\_ of this box from its parent node. Different from set it as //hidden

MMoverlay.prototype.hide = function() {}

MMoverlay.prototype.show = function() {}

MMoverlay.prototype.toggle = function() {} "hidden "and "visible" switching

MMoverlay.prototype.toggleOn = function() {} //

MMoverlay.prototype.toggleOff = function() {}

//If use right clk on map, and the cursor is not within the this.div\_, this method will set it as //"hidden"

MMoverlay.prototype.toggleOff2 = function () {} // simply toggle off,

MMoverlay.prototype.toggleDOM = function(){}

// this.setMap(this.map) will automatically run this.draw() method

MMoverlay.prototype.updatecat = function(string2) {...}

//I think I should move this method and similar ones like updatebb(string1) and updatecap\_tr to //corresponding form classes

// update value of hidden input (#num-catagory) of rental form

//update caption of drowdown button in rental form, "want to lease" "want rent" and so on

MMoverlay.prototype.redirect\_func = function(latLng, num) {...}

//

function C\_travel\_control(){

C\_travel\_control\_pointer = this;

this.\_next = 0;

this.\_myroute = null;

this.\_overview\_path = null;

this.\_overview\_path\_latlngs = []; // [ latLng1, latLng2]

this.\_jsonfied\_overview\_path\_latlngs = "";

this.marker\_array\_ = [];

this.\_tavel\_div = document.getElementById("travel-control-div");

this.\_control\_panel = document.getElementById("travel-control-panel");

this.get\_array = {"tbeid\_tr":-1,"cat\_tr": -1, "wannaget\_tr":0 };

this.\_infowindow = new google.maps.InfoWindow({ //init a infowindow

content: "default string"

});

//this.directionsDisplay = new google.maps.DirectionsRenderer(); // DirectionsRenderer is not used in this C\_travel\_control class

this.directionsService = new google.maps.DirectionsService();

}

Geocode ： humman-readable address string => google latlng object 完全是对getarray的操作，在control unit里面编写

C\_travel\_control.prototype.geocode = function ( address\_string , assoc\_array\_of\_get, ori\_or\_des ){ }

C\_travel\_control.prototype.calcRoute = function ( ) {...} 保留在control unit里面

这个方法通过获取instance的marker\_array来调用google的 directionsService获取路径，应该保留在EasySubOrg.RIDE.ride\_cu\_01

C\_travel\_control.prototype.confirmRoute= function( latLng, infostr) 将hidden inputs的填写好，以备ajax post , 这个应该也是属于control init

C\_travel\_control.prototype.hightlight\_title = function (ID) 这个暂时不管

C\_travel\_control.prototype.set\_status = function ( id, status\_num) { } 处理过时路径暂时不管

C\_travel\_control.prototype.setDefault = function() 这个函数执行相应调整 租房模式

C\_travel\_control.prototype.setTravel= function() 。。。执行调整步骤，目标为旅行模式，也就是ride模式

C\_travel\_control.prototype.updateSetting 通过对workMode变量的判断，选择对应的setDefault还是setTravel, 应该写成一个函数控制模式，然后其他函数监听workMode变量，所以setDefault setTravel可以合并

C\_travel\_control.prototype.resetGetArray = function() {this.get\_array = {"tbeid\_tr":-1,"cat\_tr": 1, "wannaget\_tr":0 };} 由 MAP.render call， 该函数写在pageinteraction, 属于EasySubOrg.RIDE.cu\_01的方法

C\_travel\_control.prototype.getAndShow = function() {} 应该写在 comm\_unit，改方法从EasySubOrg.RIDE.cu\_01获取数据，然后将服务器结果update到 MAP Control Unit， MAP render listen并呈现

原方法去除所有features, markers，custom\_routes， AJAX get 一次， mapcc1.adaptiveSetFeature

拆分 RequestSearchResult

comm\_unit相应函数启动，并update 对应的在MAP.cu\_01中的travel\_search\_result， Render监听这一变量，并实现

C\_travel\_control.prototype.clearMarkers C\_travel\_control.prototype.deleteMarkers 两兄弟函数，不多说，改动到mapinteraction

全部合并到map interaction分别只用一个函数

C\_travel\_control.prototype.addOneWaypoint改动到mapinteraction

C\_travel\_control.prototype.display\_overview\_path = function() { ....} 需要移动到mapinteraction板块

C\_travel\_control.prototype.addMarker = function(location, cat, isdraggable) { 。。。}改动到mapinteraction

这个函数对cat 有一些判断， cat ==0是起点，cat==1是终点，cat==2以及其他，则添加位标

还负责点选生成后路径的起点终点标识。

C\_travel\_control.prototype.set\_origin = function ( latLng) { }改动到mapinteraction

C\_travel\_control.prototype.set\_destination = function ( latLng ) { }改动到mapinteraction

C\_travel\_control.prototype.set\_next = function (latLng) { } // end of set\_next()

C\_travel\_control.prototype.gen\_url = function(get\_type) {} comm\_unit

C\_travel\_control.prototype.get = function( get\_type ) {} 处理两种，设置过时，或者拉去用搜索出来的路径

C\_travel\_control.prototype.post = function () { } 《- C\_travel\_control.prototype.post\_para\_gen{}

=====================================================

=====================================================

C\_search\_ajax() {

}

**C\_search\_ajax.prototype.update\_get\_array** = function (key,value ) { this.get\_array[key] = value; // 可以不用了

//alert(key + " "+ this.get\_array[key]);

}

**C\_search\_ajax.prototype.addMarker** =function(location, cat, url\_str, memo, id)

//处理cat =1

**C\_search\_ajax.prototype.setAllmap** = function(map) {

//alert("set entered");

for (var i = 0; i < this.marker\_array\_.length; i++) {

this.marker\_array\_[i].setMap(map);

}

}

**C\_search\_ajax.prototype.clearMarkers**  = function () {

//alert("clear entered");

this.setAllmap(null);

}

**C\_search\_ajax.prototype.deleteMarkers** = function() {

//alert("delete entered");

this.clearMarkers();

this.marker\_array\_ = [];

}

C\_search\_ajax.prototype.set\_status = function (id, status\_num, entry)

C\_search\_ajax.prototype.gen\_url

C\_search\_ajax.prototype.get // ajax

this.addMarker(latLng, 4,null,null,null,true);