

A world map is shown in a dark, monochromatic style. Overlaid on the map are several vertical bands of different colors: a light blue band on the left, a green band in the center, and a dark blue band on the right. The text is overlaid on these bands.

SuperMap iServer Java 课程（三）

脚本开发（1）

培训部

北京超图软件股份有限公司

主要内容



SuperMap iServer Java 层次结构

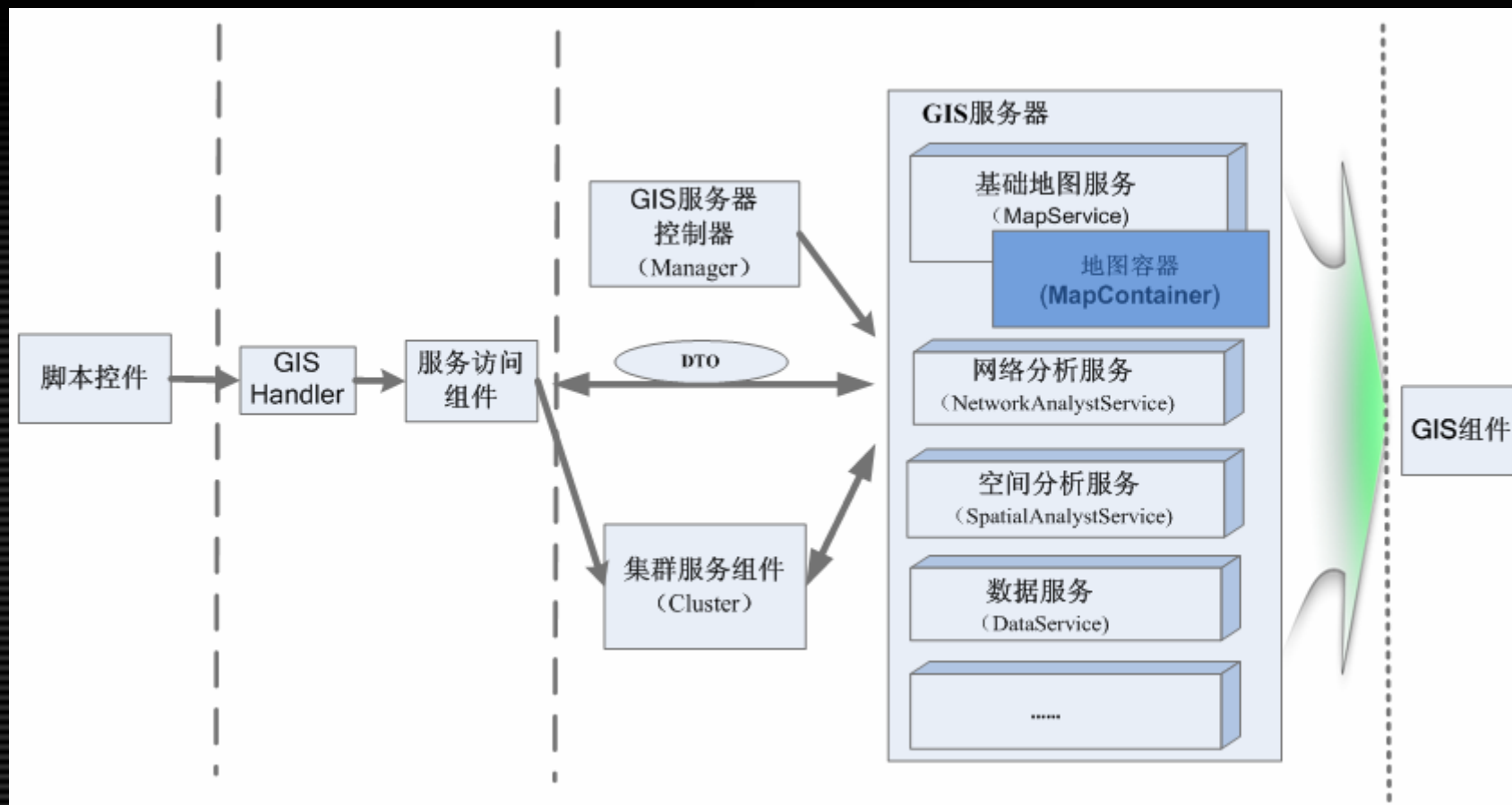


开发方式



二次开发

1、SuperMap iServer Java 层次结构



2、开发方式



客户端

基于模板开发

使用 JavaScript SDK 开发

Web服务层

使用 Web 控件开发

扩展 Map 开发

GIS服务器层

扩展 MapContainer 开发

3、二次开发



- 地图的基本浏览功能
- 地图量算
- 空间查询
- 属性查询
- 缓冲分析
- 最近地物查找
- 鹰眼功能

3.1 地图的基本浏览功能



1、全图显示

```
mapControl.viewEntire();
```

2、快速放大

```
mapControl.zoomIn();
```

3、拉框放大

```
var zoomInAction = new SuperMap.UI.ZoomInAction();  
mapControl.set_action(zoomInAction);
```

3.1 地图的基本浏览功能



4、快速缩小

```
mapControl.zoomOut();
```

5、拉框缩小

```
var zoomOutAction = new SuperMap.UI.ZoomOutAction();  
mapControl.set_action(zoomOutAction);
```

6、平移

```
var panAction = new SuperMap.UI.PanAction();  
mapControl.set_action(panAction);
```

3.2 地图量算



1、距离量算

```
var measureDistanceAction =  
new SuperMap.UIT.MeasureDistanceAction(onMeasureDistanceComplete, onError);  
mapControl.set_action(measureDistanceAction);
```

```
function onMeasureDistanceComplete(result){  
    var unitValue = mapControl.getMap()._coordsSys.units;  
    var temp = result.distance + getUnits(unitValue);  
    alert(temp);  
};
```

3.2 地图量算



2、面积量算

```
var measureAreaAction =  
new SuperMap.UI.MeasureAreaAction(onMeasureAreaComplete, onError);  
mapControl.set_action(measureAreaAction);
```

```
function onMeasureAreaComplete(result){  
    var unitValue = mapControl.getMap()._coordsSys.units;  
    var temp = result.area + "平方" + getUnits(unitValue);  
    alert(temp);  
};
```

3.2 地图量算



```
// 用于获取错误信息的面积量算事件处理函数  
function onError(responseText) {  
    alert(responseText);  
};
```

```
function getUnits(unitValue){  
    var units = new SuperMap.Units();  
    if(unitValue == units.meter) {  
        return "米"; } else if(unitValue == units.kilometer){  
        return "千米"; } else if(unitValue == units.mile){  
        return "英里"; } else if(unitValue == units.kilometer){  
        return "公里"; } else if(unitValue == units.yard){  
        return "码"; } else if(unitValue == units.degree){  
        return "度"; } else if(unitValue == units.millimeter){  
        return "毫米"; } else if(unitValue == units.centimeter){  
        return "分米"; } else if(unitValue == units.inch){  
        return "英寸"; } else if(unitValue == units.decimeter){  
        return "分米"; } else if(unitValue == units.foot){  
        return "英尺"; }  
    return "米"; }.
```

3.3 空间查询



- 参数:

**PointQueryAction, LineQueryAction ,
RectQueryAction, CircleQueryAction,
PolygonQueryAction**

- 方式

点选查询, 线选查询, 框选查询, 圆选查询,
多边形选查询

3.3.1点选查询

方法: **PointQueryAction**

参数:

- layerNames 图层名称
- sqlParam 过滤条件
- Tolerance 查询容限
- needHighlight 是否高亮
- onComplete 回调函数
- onError 捕错函数

3.3.1点选查询



pointQueryAction.queryParam.returnResultSetInfo 返回结果类型

(1)返回属性信息

```
= new SuperMap.ReturnResultSetInfo().returnAttribute;
```

(2)返回空间信息

```
= new SuperMap.ReturnResultSetInfo().returnGeometry;
```

(3)返回空间信息和属性信息

```
= new SuperMap.ReturnResultSetInfo().returnAttributeAndGeometry;
```

3.3.1点选查询

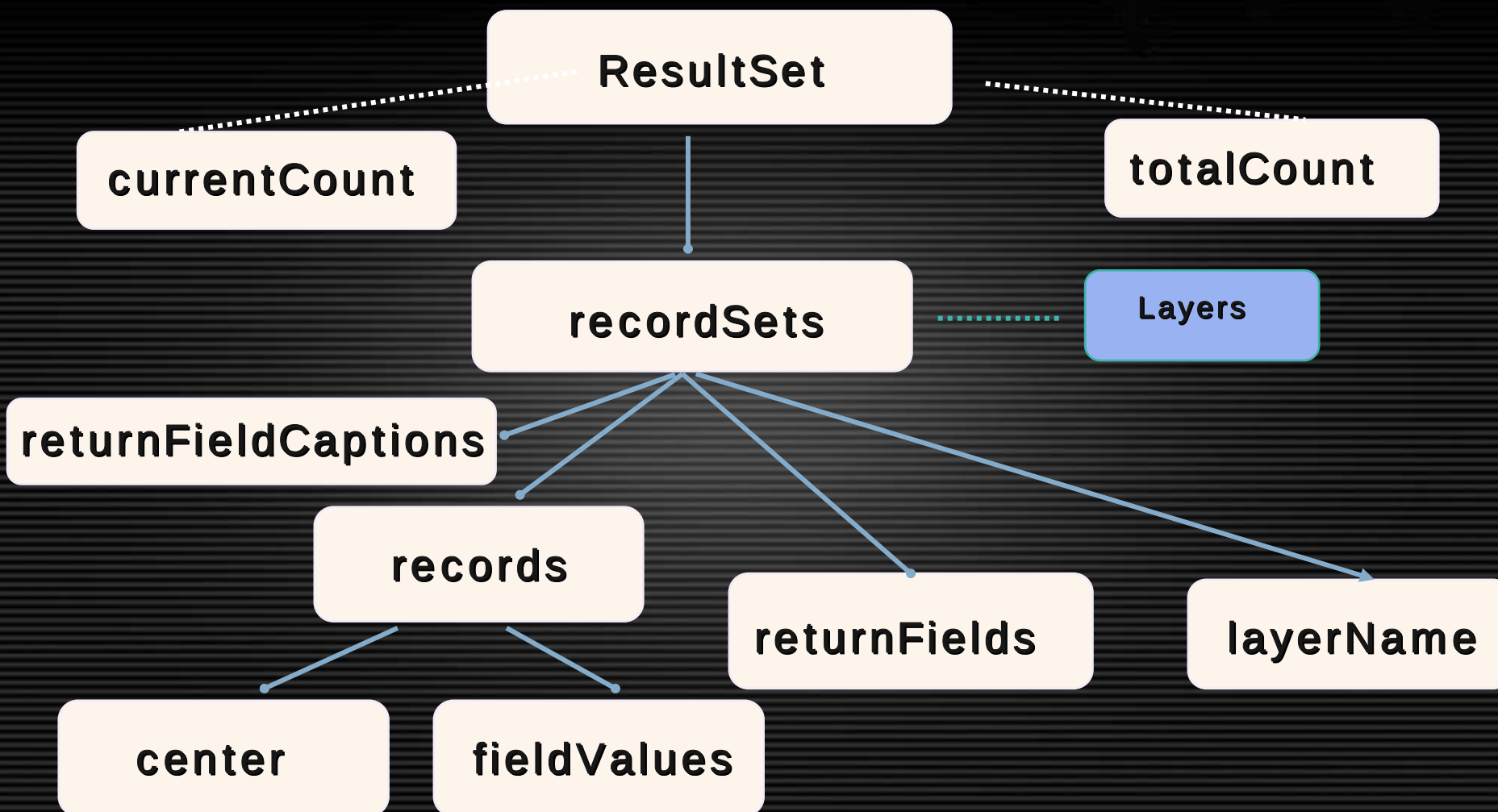


样例:

```
function setPointQuery(){
    var queryLayerNames = new Array();
    queryLayerNames[0]="School@changchun";
    var sqlParam = new SuperMap.SqlParam();
    sqlParam.whereClause ="SmID>1";
    var pointQueryAction = new SuperMap.UI.PointQueryAction(
        queryLayerNames, sqlParam, 500,true, onQueryComplete, onError);
    mapControl.set_action(pointQueryAction);
    pointQueryAction.queryParam.returnResultSetInfo =
        new SuperMap.ReturnResultSetInfo().returnAttributeAndGeometry;
}
```

3.3.1 点选查询

— 查询结果的解析



3.3.2其他查询



方法:	LineQueryAction	线选查询
	RectQueryAction	框选查询
	CircleQueryAction	圆选查询
	PolygonQueryAction	多边形选查询

参数:

- layerNames 图层名称
- sqlParam 过滤条件
- needHighlight 是否高亮
- onComplete 回调函数
- onError 捕错函数

3.4 属性查询



- 属性查询
- 查询条件的设置
- 查询结果的解析

3.4属性查询

方法: `map.queryBySql(...)`

参数:

- queryParam 查询参数
- needHighlight 是否高亮
- onComplete 回调函数
- onError 捕错函数

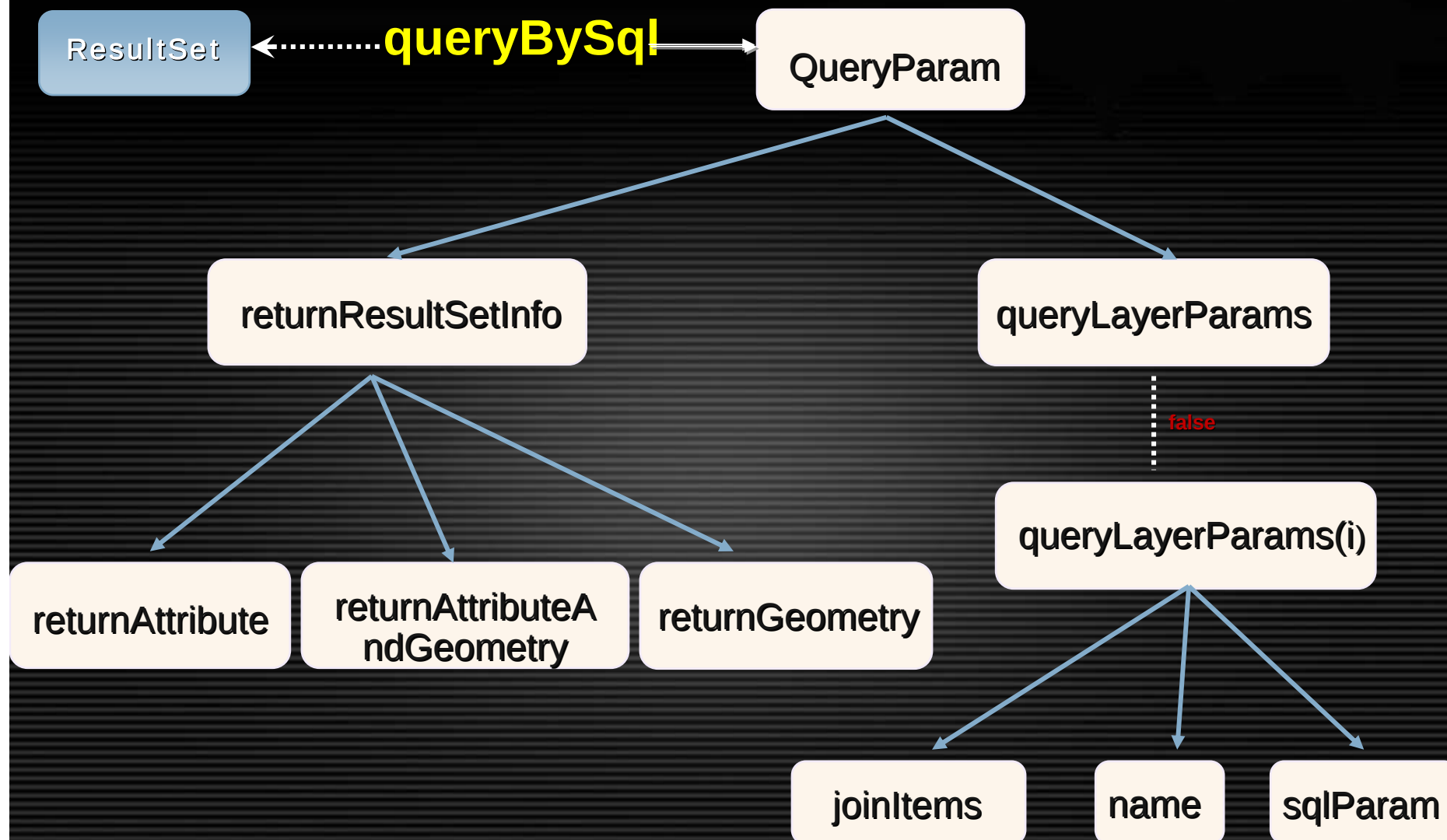
3.4.1 属性查询



样例:

```
function sqlQuery() {  
    var queryParam = new SuperMap.QueryParam();  
    var sqlParam = new SuperMap.SqlParam();  
    sqlParam.whereClause = "SmID>1";  
    queryParam.queryLayerParams = new Array();  
    var queryLayerParam = new SuperMap.QueryLayerParam();  
    queryLayerParam.name = "School@changchun";  
    queryLayerParam.sqlParam = sqlParam;  
    queryParam.queryLayerParams.push(queryLayerParam);  
    queryParam.expectCount = 100;  
    mapControl.getMap().queryBySql(queryParam, true,  
        onQueryComplete, onError);  
}
```

3.4.3 查询条件的设置



3.4.4 查询结果的解析



`onQueryComplete( resultSet );`

`resultSet`



3.5 缓冲分析



- 参数:

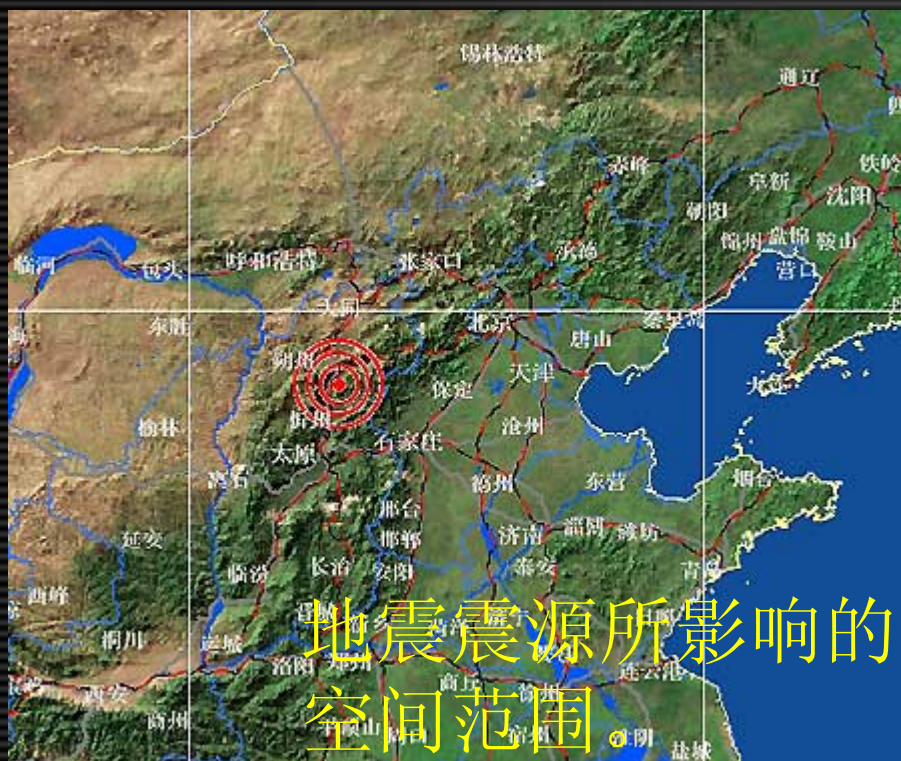
PointBufferQueryAction, LineBufferQueryAction
RectBufferQueryAction, CircleBufferQueryAction
PolygonBufferQueryAction

- 方式:

点选缓冲分析, 线缓冲分析,
框选缓冲分析, 圆选缓冲分析,
多边形选缓冲分析

3.5 缓冲分析

应用说明：根据指定的距离在点、线和多边形实体周围自动建立一定宽度的区域范围的分析方法。例如，在环境治理时，常在污染的河流周围划出一定宽度的范围表示受到污染的区域；又如在飞机场，常根据健康需要在周围划分出一定范围的区域作为非居住区，等等。



3.5.1 点选缓冲分析

方法: **PointBufferQueryAction**

参数:

- **layerNames** 图层名称
- **sqlParam** 过滤条件
- **bufferAnalystParam** 设置参数
- **queryMode** **(-1,0,1,2,3,5,4,6,7,8)** 查询模式
- **needHighlight** 是否高亮
- **onComplete** 回调函数
- **onError** 捕错函数

3.5.1点选缓冲分析



样例:

```
function setPointBufferQuery() {  
    var queryLayerNames = new Array();  
    queryLayerNames[0] = "School@changchun";  
    var bufferAnalystParam = new SuperMap.BufferAnalystParam();  
    bufferAnalystParam.bufferEndType = 0;  
    bufferAnalystParam.leftDistance = "1000";  
    bufferAnalystParam.semicircleLineSegment = "10";  
    var queryMode = "2"; //返回与搜索对象相交的所有对象  
    var pointBufferQueryAction = new SuperMap.UI.PointBufferQueryAction(  
        queryLayerNames, null, bufferAnalystParam, queryMode, true,  
        onQueryComplete, onError);  
    mapControl.set_action(pointBufferQueryAction);  
}
```

3.5.1点选缓冲分析



- 缓冲分析结果解析

resultSet



3.5.2 其他缓冲分析



方法: **RectBufferQueryAction** 框选缓冲分析

LineBufferQueryAction 线选缓冲分析

PolygonBufferQueryAction 多边形选缓冲分析

参数:

- | | |
|---|------|
| ■ layerNames | 图层名称 |
| ■ sqlParam | 过滤条件 |
| ■ bufferAnalystParam | 设置参数 |
| ■ queryMode (-1,0,1,2,3,5,4,6,7,8) | 查询模式 |
| ■ needHighlight | 是否高亮 |
| ■ onComplete | 回调函数 |
| ■ onError | 捕错函数 |

3.6 最近地物查找

方法: **FindNearestAction**

参数:

- | | |
|-----------------|------|
| ■ layerNames | 图层名称 |
| ■ sqlParam | 过滤条件 |
| ■ tolerance | 查询容限 |
| ■ needHighlight | 是否高亮 |
| ■ onComplete | 回调函数 |
| ■ onError | 捕错函数 |

3.6 最近地物查找



样例:

```
function setFindNearest(){  
    var layerNames = new Array();  
    layerNames[0] = "Hospital@changchun";  
    var sqlParam = new SuperMap.SqlParam();  
    var tolerance = 1000;  
    var findNearestAction = new SuperMap.UI.FindNearestAction(  
        layerNames,sqlParam,tolerance, true, onQueryComplete, onError);  
    mapControl.set_action(findNearestAction);  
}
```

3.6 最近地物查找



- 最近地物查找结果解析

resultSet



3.7 鹰眼功能



鹰眼功能: **OverviewControl**控件

OverviewControl常用属性:

map : 获取或设置与鹰眼相关联的**MapControl**对象
(**get_map** 或**set_map**)

Param: 设置鹰眼控件相关的参数, 如索引框边线颜色, 风格宽度等(**get_param** 或**set_param**)

3.7 鹰眼功能



■ 样例:

```
function Overview(){  
    var overviewDiv = document.getElementById("view");//"view"为div的id  
    overviewControl = new SuperMap.UI.OverviewControl(overviewDiv);  
    overviewControl.set_map(mapControl);  
    overviewControl.set_param(null);  
    overviewControl.initialize();  
};
```

A world map is shown in a dark, monochromatic style with a greenish-blue tint. The map is overlaid with several vertical bands of varying colors, including shades of green, blue, and purple, which create a sense of depth and movement. The text "Thank You !" is centered over the map in a large, bold, white font with a dark purple outline.

Thank You !

www.supermap.com.cn