# EChart

## 基础

纯js图表库；

给sublime text 3装好插件后，你就可以把“echart官网”上的第一个例子复制下来运行了；

## Chrome调试

博客html；

## 名词

series 一份数据，series.data 数据项 ；

series.data [ [12,54,’梵蒂冈’], //三维数据项，如x,y,气泡半径；

## 图表

### 饼图

myChart.setOption({

series : [

{

name: '访问来源',

type: 'pie',

radius: '55%',

data:[

{value:235, name:'视频广告'},

{value:274, name:'联盟广告'},

{value:310, name:'邮件营销'},

{value:335, name:'直接访问'},

{value:400, name:'搜索引擎'}

]

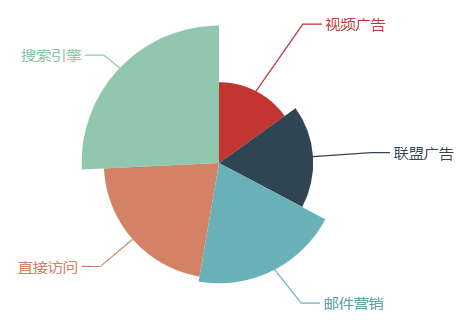
}

]

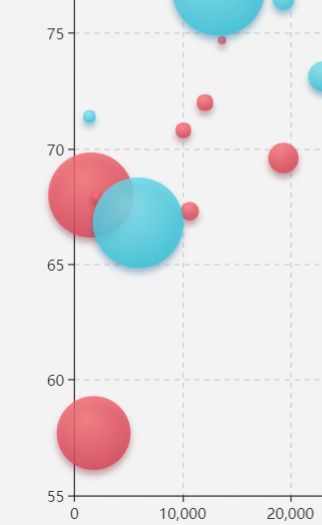
})

### 南丁格尔图

饼图加一句：roseType: 'angle'



### 气泡图



option = {

yAxis: { scale: true去掉y轴的0刻度 },

series: [ {

//多维度数据:[GDP,寿命,气泡大小,tooltip,分类]

data: [28604,77,17096869,'Australia',1990]

symbolSize: function(d){ //d就是data，数组或者值

return Math.sqrt(d[2]) / 5e2; //气泡大小

},

label: { emphasis: { show: true,

formatter: function(param){ //鼠标悬停tooltip

return param.data[3]; } } },

itemStyle:{ normal: { 有阴影好看,

//参考itemStyle.normal.color有线性渐变、径向渐变

color: new echarts.graphic.RadialGradient(0.4, 0.3, 1, [{ //前3个参数是圆心x,y,半径r, 0-1,图形占比

offset: 0, //0%处颜色

color: 'rgb(251, 118, 123)'

}, {

offset: 1,//100%处颜色

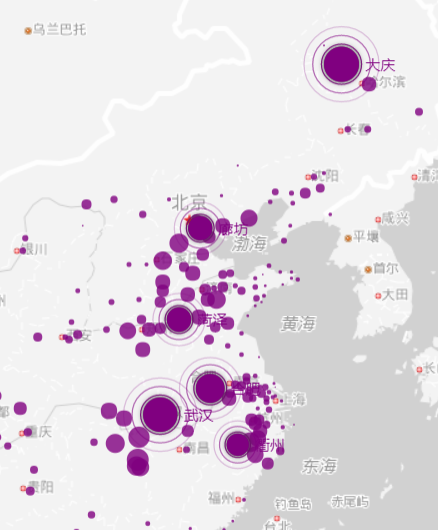
color: 'rgb(204, 46, 72)'

}])

} }

}, {} ]

}；



var data = [ {name:’海门’, value: 9 },... ]; //pm2.5值

var geoCoordMap = { ‘海门’:[121.15, 31.89],... };//城市经纬度

var convertData = function (data) {

var res = [];

for (var i = 0; i < data.length; i++) {

var geoCoord = geoCoordMap[data[i].name]; //[经纬度]

if (geoCoord) {

res.push({

name: data[i].name,

value: geoCoord.concat(data[i].value) //[经纬度,pm2.5]拼接json数组

});

}

}

return res;

};

series:[

{

name: ‘pm2.5’,

coordinateSystem: ‘bmp’, //自定义坐标系，散点图实例

},

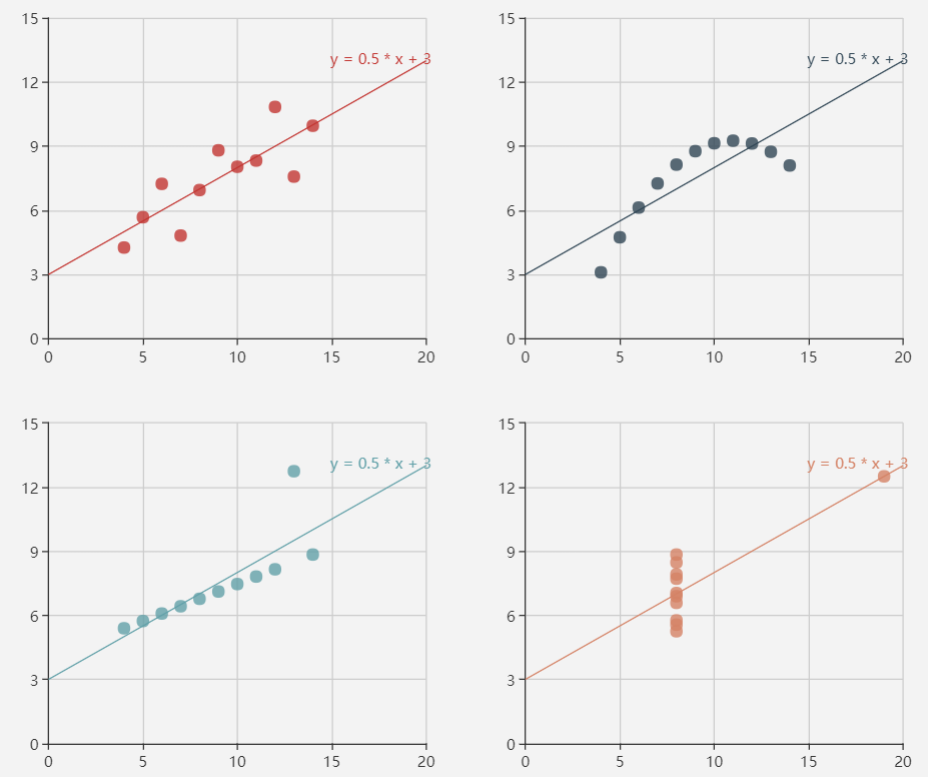
{

name:’Top 5’,

data: data.sort(function(a,b){ return b.value – a.value; }).slice(0,5)//取topN

}

]



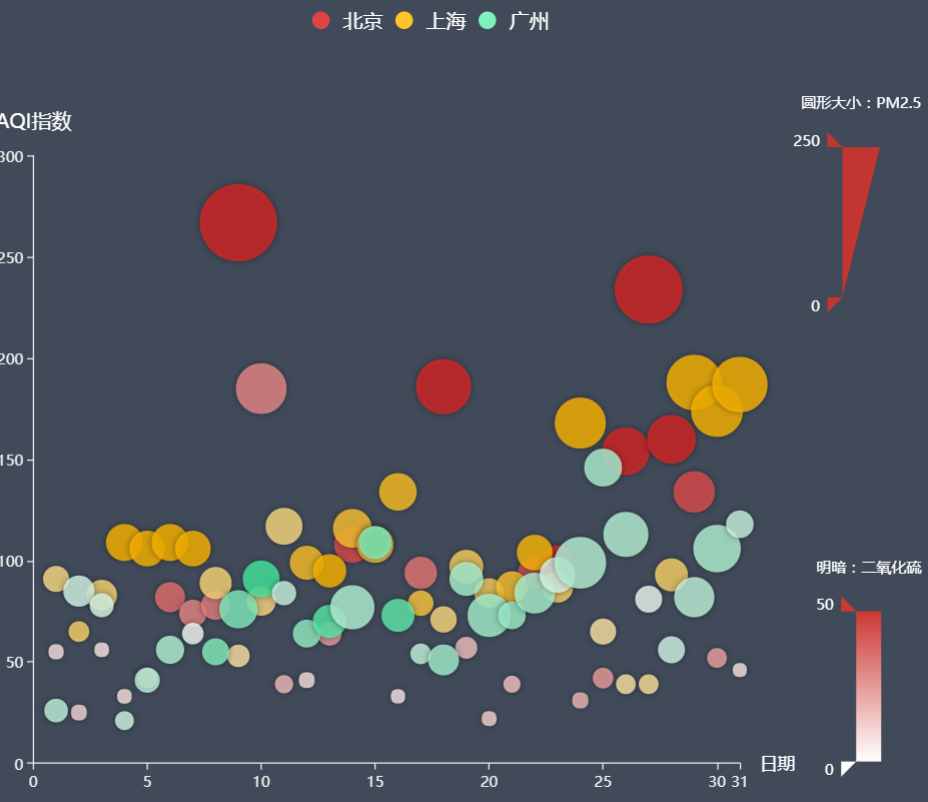
option = {

grid: [ {x: '7%', y: '7%', width: '38%', height: '38%'},... ], //单个echart实例多个grid组件

xAxis: [ { gridIndex:0,...} ],

markLine: ... //见实例，标线

};



var dataBJ = [

[1,55,9,56,0.46,18,6,"良"], ... ]; //多维度：日期x轴、AQIindex y轴、PM2.5 圆形大小、二氧化硫明暗度

var schema = [ //数据维度,tooltip使用

{name: 'date', index: 0, text: '日'},

{name: 'AQIindex', index: 1, text: 'AQI指数'},

{name: 'PM25', index: 2, text: 'PM2.5'},

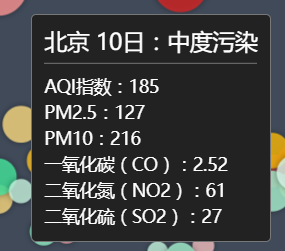
{name: 'PM10', index: 3, text: 'PM10'},

{name: 'CO', index: 4, text: '一氧化碳（CO）'},

{name: 'NO2', index: 5, text: '二氧化氮（NO2）'},

{name: 'SO2', index: 6, text: '二氧化硫（SO2）'}

];



option = {

tooltip: {

padding: 10,

backgroundColor: '#222',

borderColor: '#777',

borderWidth: 1,

formatter: function (obj) { //自定义tooltip

var value = obj.value;

return '<div style="border-bottom: 1px solid rgba(255,255,255,.3); font-size: 18px;padding-bottom: 7px;margin-bottom: 7px">'

+ obj.seriesName + ' ' + value[0] + '日：'

+ value[7]

+ '</div>'

+ schema[1].text + '：' + value[1] + '<br>'

+ schema[2].text + '：' + value[2] + '<br>'

+ schema[3].text + '：' + value[3] + '<br>'

+ schema[4].text + '：' + value[4] + '<br>'

+ schema[5].text + '：' + value[5] + '<br>'

+ schema[6].text + '：' + value[6] + '<br>';

}

}

};

visualMap映射，实例；

### 阴影&透明度&颜色&边框

#### 一般都会在系列(series)的 [itemStyle](http://echarts.baidu.com/tutorial.html#series-pie.itemStyle) 里设置；

itemStyle: {

normal: {

shadowBlur: 200,// 阴影的大小

shadowOffsetX: 0, // 阴影水平方向上的偏移

shadowOffsetY: 0,// 阴影垂直方向上的偏移

shadowColor: 'rgba(0, 0, 0, 0.5)',// 阴影颜色

color:’#c23532’//颜色

},

// emphasis: { hover显示阴影

// shadowBlur: 200,

// shadowColor: 'rgba(0, 0, 0, 0.5)'

// }

}

#### 为每个数据设置颜色（嵌套思想）

data: [{

value:400,

name:'搜索引擎',

itemStyle: {

normal: {

color: '#c23531' //ctrl+shift+c

}

}

}, ...]

### 背景色&文本色

背景、文本等颜色可以全局可以局部；

setOption({

backgroundColor: '#2c343c',

textStyle: {

color: 'rgba(255, 255, 255, 0.3)'

}

})

单独设置：series.[label.normal.textStyle](http://echarts.baidu.com/option.html#series-pie.label.normal.textStyle) 。。。

### 引导线

series下：

labelLine:{

normal:{

lineStyle:{color:'rgba(255,255,255,0.3)'}

}

}

### 将数值大小映射为明暗度的变化

visualMap: {

show: false,// 不显示 visualMap 组件，只用于明暗度的映射

min: 80, // 映射的最小值为 80

max: 600,// 映射的最大值为 600

inRange: {

colorLightness: [0, 1]// 明暗度的范围是 0 到 1

}

},

backgroundColor:'#2c343c',

series : [

//data不要给颜色

//itemStyle.normal.color给一种颜色，按它改变明暗度

### 异步数据刷新

myEChart-异步获取数据.html：1）初始化option（除了数据）；2）数据来了，调用函数显示数据（匹配series.name）；

### 提示loading

myChart.showLoading(); //初始化option

myChart.hideLoading(); //数据来了

### 数据

#### js. shift

删除数组第一个元素，data.push(...)，data.shift(); 保证数据不变；

### 交互组件

图例组件 [legend](http://echarts.baidu.com/option.html#legend)、标题组件 [title](http://echarts.baidu.com/option.html#title)、视觉映射组件 [visualMap](http://echarts.baidu.com/option.html#visualMap)、数据区域缩放组件 [dataZoom](http://echarts.baidu.com/option.html#dataZoom)、时间线组件 [timeline](http://echarts.baidu.com/option.html#timeline)

#### [dataZoom](http://echarts.baidu.com/option.html#dataZoom)

一个数轴一个；

窗口范围目前支持：百分比 start，和绝对数值 startValue；

子组件：dataZoomInside内置型、Slider滑动条、Select框选；

option = {

xAxis: {

type: 'value'

},

yAxis: {

type: 'value'

},

dataZoom: [

{ // 这个dataZoom组件，默认控制x轴。

type: 'slider/inside',

start: 10, // 左边在 10% 的位置。

end: 60 // 右边在 60% 的位置。

yAxisIndex: 0

}

},

series:[]

}

#### [visualMap](http://echarts.baidu.com/option.html#visualMap)

折线图把数据映射到『线』，柱状图把数据映射到『长度』，echart每种图表都内置了映射过程；

visualMap提供了通用的视觉映射，视觉元素有：  
图形类别（symbol）、图形大小（symbolSize）  
颜色（color）、透明度（opacity）、颜色透明度（colorAlpha）、  
颜色明暗度（colorLightness）、颜色饱和度（colorSaturation）、色调（colorHue）

一般的图表，默认把一维映射到x，二维映射到y，气泡图这种三维映射到半径，visualMap可以定义把“哪个维度”映射到“什么视觉元素”上；

如：

世界人口分布（人口数量 + 颜色）；

主要城市空气质量（城市 + pm2.5值所属范围）

option = {

visualMap: [

{

type: 'piecewise'

min: 0,

max: 5000,

dimension: 3, // series.data 的第四个维度（即 value[3]）被映射

seriesIndex: 4, // 对第四个系列进行映射。

inRange: { // 选中范围中的视觉配置

color: ['blue', '#121122', 'red'], // 定义了图形颜色映射的颜色列表，

// 数据最小值映射到'blue'上，最大值映射到'red'上，其余自动线性计算。

symbolSize: [30, 100] // 定义了图形尺寸的映射范围，

// 数据最小值映射到30上，最大值映射到100，其余自动线性计算。

},

outOfRange: { // 选中范围外的视觉配置

symbolSize: [30, 100]

}

},

...

]

};

### 事件

对应DOM事件名称，echart中分两种：1）鼠标（点击、hover等）；2）使用组件触发的，如’datazoom’事件，’legendselected’事件等；

'click'、'dblclick'、'mousedown'、'mousemove'、'mouseup'、'mouseover'、'mouseout'

// 处理点击事件并且跳转到相应的百度搜索页面

myChart.on('click', function (params) {

window.open('https://www.baidu.com/s?wd=' + encodeURIComponent(params.name));

});

params

所有的鼠标事件包含参数 **params**，这是一个包含点击图形的数据信息的对象，格式见官网-》教程-》echarts事件和行为；

判断鼠标点击在哪里；

// 通过饼图表现单个柱子中的数据分布

**组件事件**

events手册；

// 图例开关的行为只会触发 legendselectchanged 事件

myChart.on('legendselectchanged', function (params) {

//...

**代码触发echarts组件行为**

action手册；

在 ECharts 3 里改为通过调用 myChart.dispatchAction({ type: '' }) 触发图表行为；

### 拖拽

教程

### 日历坐标系

每一天的情况，图表 + 坐标系， 很强势！

### 自定义series类型

教程

### 富文本标签

教程

## IWebBrowser2与C++交互

### c++调js

# MSChart使用

## 添加MSChart

工具-》COM组件（.NET 3.5那个是给winform、web用的）或者右键添加ActiveX；

如果本地没有，就去网上下载MSChart.exe/dll， MSChart20.OCX，先安装exe，再注册ocx，win7可能要“管理员方式注册”；

添加类成员

CMschart2 m\_demoChart; 这个名字可以取好点；

## 属性介绍

Anotations 注解，箭头云朵矩形...颜色字体大小...

ChartAreas 区域，一张图XY数据...

Legends 图例, 标注图形中各个线条或颜色的含义

Series 数据

其他属性：标题、标签、tooltip、url...

ChartAreas：绘图区域，独立的图表组、数据源。

AlignmentOrientation：两个绘图区域间的对齐方式。

AlignmentStyle：图表间用以对其的元素。  
 AlignWithChartArea：参照对齐的绘图区名称。?  
        InnerPlotPosition：图表在绘图区内的位置属性。  
        Auto：是否自动对齐。  
        Height：图表在绘图区内的高度（百分比，取值在 0-100  ）  
        Width：图表在绘图区内的宽度（百分比，取值在 0-100  ）  
        X,Y：图表在绘图区内左上角坐标  
        Position：绘图区位置属性，同InnerPlotPosition。  
        Name：绘图区名称。  
        Axis：坐标轴集合  
        Title：坐标轴标题  
        TitleAlignment：坐标轴标题对齐方式  
        Interval：轴刻度间隔大小  
        IntervalOffset：轴刻度偏移量大小  
        MinorGrid：次要辅助线  
        MinorTickMark：次要刻度线  
        MajorGrid：主要辅助线  
        MajorTickMark：主要刻度线  
        DataSourceID：MSChart的数据源。  
        Legends：图例说明。  
        Palette：图表外观定义。  
        IsValueShownAsLabel：是否显示数据点标签，如果为 true，在图表中显示每一个数据值  
        Label：数据点标签文本  
        LabelFormat：数据点标签文本格式  
        LabelAngle：标签字体角度  
        Name：图表名称  
        Points：数据点集合  
        XValueType：横坐标轴类型

## CMschart2类整理

### 创建：

Create

m\_Chart.Create("mschart", WS\_CHILD| WS\_VISIBLE, rc, this, 10))

Create(序列化...)

### 错误：

枚举 ErrorConstants { VtErrorCorruptData...}

### 拖拽：

枚举 OLEDropConstants OLEDragConstants；

### 对齐：

枚举 VtHorizontalAlignment VtVerticalAlignment

VtOrientation

### 角度：

枚举 VtAngleUnits

### 打印：

枚举 VtPrintScaleType VtPrintOrientation

### 字体：

枚举 VtFontStyle VtFontEffect

### 画刷：

枚举 VtBrushStyle VtBrushPatterns VtBrushHatches

### 阴影：

枚举 VtShadowStyle

### 画笔：

枚举 VtPenStyle VtPenJoin VtPenCap

### Frame：

枚举 VtFrameStyle

### 图：

枚举 VtPictureType VtPictureOption

VtPictureMapType VtGradientStyle渐变

VtFillStyle VtMarkerStyle标记

VtProjectionType类型 VtSmoothingType VtDcType

VtChChartType图表类型

VtChSeriesType数据类型

VtChPartType 部件类型：标题轴数据...

VtChLocationType 位置类型：上下左右...

VtChLabelComponent 标签类型 VtChLabelLineStyle VtChLabelLocationType VtChSubPlotLabelLocationType

VtChPieWeightBasis VtChPieWeightStyle 饼图

VtChAxisId 轴

VtChDateIntervalType 数据间隔

VtChScaleType 刻度

VtChPercentAxisBasis

VtChAxisTickStyle

VtChStats 统计量

VtChUpdateFlags 更新标识

VtChMouseFlag鼠标

VtChSsLinkMode

搜索关键词查找；

方法：直接看demo；

## 其他

### 拷贝数组

COleSafeArray saRet;

DWORD numElements[] = {10, 10}; // 10x10

// Create the safe-array...

saRet.Create(VT\_R8, 2, numElements);

// Initialize it with values...

long index[2];

for(index[0]=0; index[0]&lt;10; index[0]++) {

for(index[1]=0; index[1]&lt;10; index[1]++) {

double val = index[0] + index[1]\*10;

saRet.PutElement(index, &val);

}

}

// Return the safe-array encapsulated in a VARIANT...

m\_Chart.SetChartData(saRet.Detach());

m\_Chart.Refresh;

### demo

codeproject：搜索 MFC MSChart

使用：

# Excel 自动化

## 其他

### msdn接口说明

excel：<https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel.aspx>

application：<https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel.application_members.aspx>

workbooks：[workbooks\_members.aspx](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel.workbooks_members.aspx)

workbook：workbook.aspx

sheets：sheets\_members.aspx

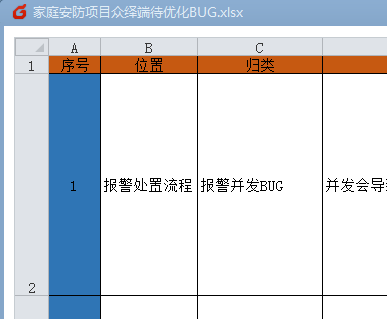
windows：windows\_members.aspx

window：window\_members.aspx

chart：chart\_members.aspx

chartgroup：chartgroups.aspx

### Foxmail就是内嵌excel的：



打开excel文件时，任务管理器多出一个excel.exe，打开多个excel文件，共享一个进程；

### 问题记录

2，不保存界面状态

3，跟随对话框移动

4，右键菜单取消

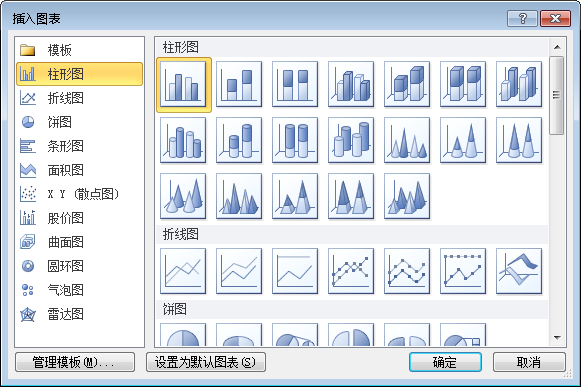
5，

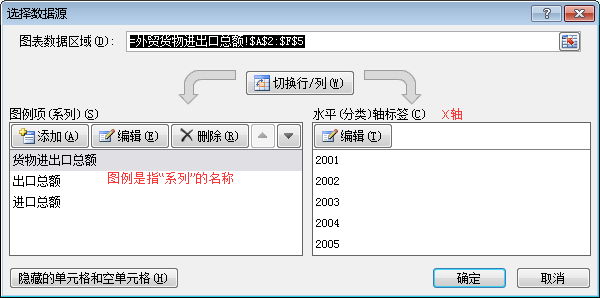
### Sheets 和 Worksheets的区别

Sheets集合代表当前工作簿中的所有工作表,包括图表工作表、对话框工作表和宏表。

Worksheets集合仅代表当前工作簿中的所有工作表。

## 统计图表制作



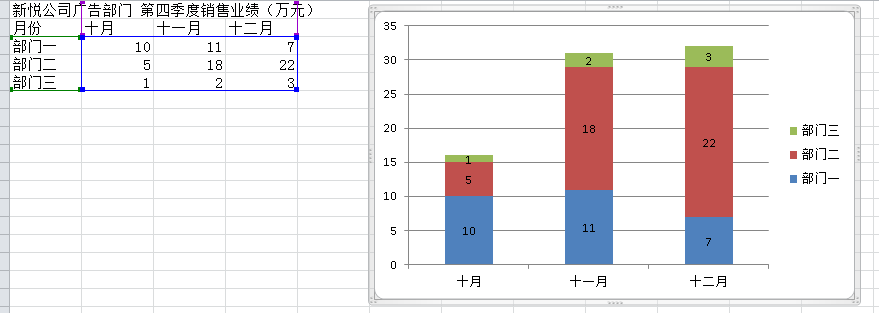


### 问题记录

* 一种数据源无法用多种统计方式显示在一个绘图区域内；

### 柱形图

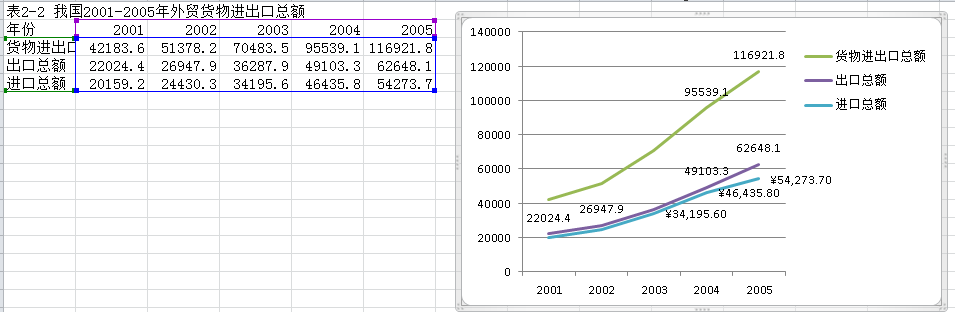
显示或比较多个数据组；



右键各“部件”，可以设置很多东西，做出来的图也不一样，有“网格线

数据点（一条柱子），绘图区域，标签，图例，坐标轴。。。”；

### 折线图



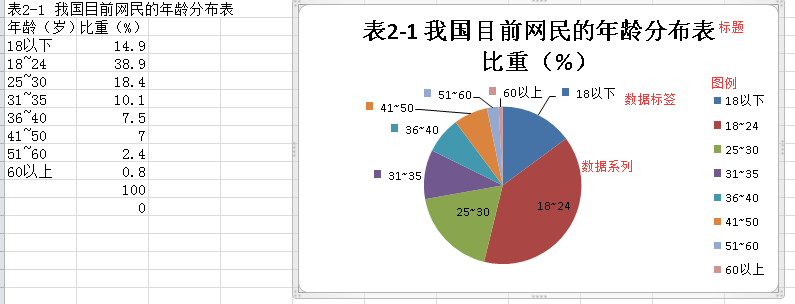
### 饼图

**整体和部分**的关系，常用在研究结构性问题；

输入数据： 《excel表格模板，网民的年龄分布表》，“插入-》图表”，选择“饼图”

右键空白的画布，添加数据源，拖选单元格，设置“标题、坐标轴、图例。。。”；

或者先选定数据源：“标题，图例，系列数据，标签。。。”，右键添加标签及设置格式；



### 条形图

类似柱状图

## Chart接口

### 2-D图表属性

[ChartArea](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.chartarea.aspx) 图表区域

[ChartStyle](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.chartstyle.aspx) 图表风格

[ChartTitle](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.charttitle.aspx) 图表标题

[HasTitle](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.hastitle.aspx)

[ChartType](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.charttype.aspx) 图表类型

[PlotArea](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.plotarea.aspx) 绘图区域

[PlotBy](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.plotby.aspx)  [XlRowCol](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel.xlrowcol.aspx)常量，指定行列哪个用于data series

[PlotVisibleOnly](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.plotvisibleonly.aspx) 只绘制可见cells

[DataTable](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.datatable.aspx) 图表的数据源

[HasDataTable](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.hasdatatable.aspx)

[DisplayBlanksAs](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.displayblanksas.aspx) 空白处理

[ProtectContents](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.protectcontents.aspx) 图表不可修改

[ProtectData](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.protectdata.aspx) data series不可修改

[ProtectDrawingObjects](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.protectdrawingobjects.aspx) 形状不可修改。。。

[SizeWithWindow](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.sizewithwindow.aspx) 图表适应窗口

[Visible](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.visible.aspx) 可见性

[HasAxis](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.hasaxis.aspx) 坐标轴

[Legend](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.legend.aspx) 图例

[HasLegend](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.haslegend.aspx)

### 方法

[ApplyCustomType](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.applycustomtype.aspx) 应用标准或自定义图表类型

https://msdn.microsoft.com/en-us/library/bb241008(v=office.12).aspx

[ApplyDataLabels](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.applydatalabels.aspx) 应用data label

[AreaGroups](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.areagroups.aspx) 返回一个[ChartGroup](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel.chartgroup.aspx)或[ChartGroup](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel.chartgroup.aspx)s对象

[ColumnGroups](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.columngroups.aspx) 柱形

[Axes](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.axes.aspx) 返回坐标轴

[LineGroups](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.linegroups.aspx) 折线

[PieGroups](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.piegroups.aspx) 饼

[BarGroups](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.bargroups.aspx) 条形

[RadarGroups](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.radargroups.aspx) 雷达

[XYGroups](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.xygroups.aspx) 散列

[ChartGroups](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.chartgroups.aspx) 所有，包括Area，Bar。。。

[ChartObjects](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.chartobjects.aspx) embedded chart或集合

[ChartWizard](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.chartwizard.aspx) 修改chart属性

[Location](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.location.aspx) 移动chart

[SeriesCollection](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.seriescollection.aspx) 数据

[SetBackgroundPicture](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.setbackgroundpicture.aspx) 背景图片

[SetElement](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.setelement.aspx) 设置chart元素

[SetSourceData](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.setsourcedata.aspx) 设置chart的数据range

[Unprotect](https://msdn.microsoft.com/en-us/library/office/microsoft.office.interop.excel._chart.unprotect.aspx) 取消保护

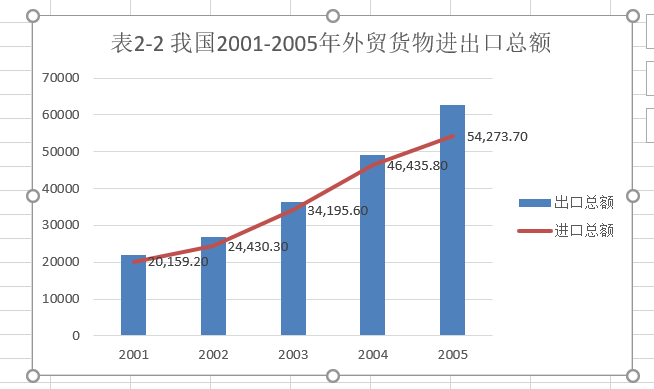
#### chartgroup

一个chart有1-n个chartgroup；

每个chartgroup有1-n个series；

每个series有1-n个points；

下面就是一个chart，它有两个chartgroup，每个chartgroup有1个series，一个series有5个点；



打印类、3-D类。。。

### 控制excel可见

app.put\_Visible(TRUE);

app.put\_UserControl(TRUE); //用户控制创建

ribbon菜单-》视图，显示那里可以控制，网格、编辑框、标题的显示；

代码中，workbook打开后可以设置app.visible = FALSE;

### 设置excel位置

知识点：

字体以points为单位时，1point = 1/72 英寸；

平时我们用的显示器分辨率是：96DPI（dots-per-inch点每英寸），对于高DPI如120,144,192，windows处理ui的时候分别放大100%，125%，150%，200%，详见《html windows高DPI...》；

实际计算：

默认windows都假定96DPI，

points = 3/4 \* pixels；

将excel放到dialog客户区：

app.put\_Left(rc.left \* 3 / 4);

app.put\_Top(rc.top \* 3 / 4);

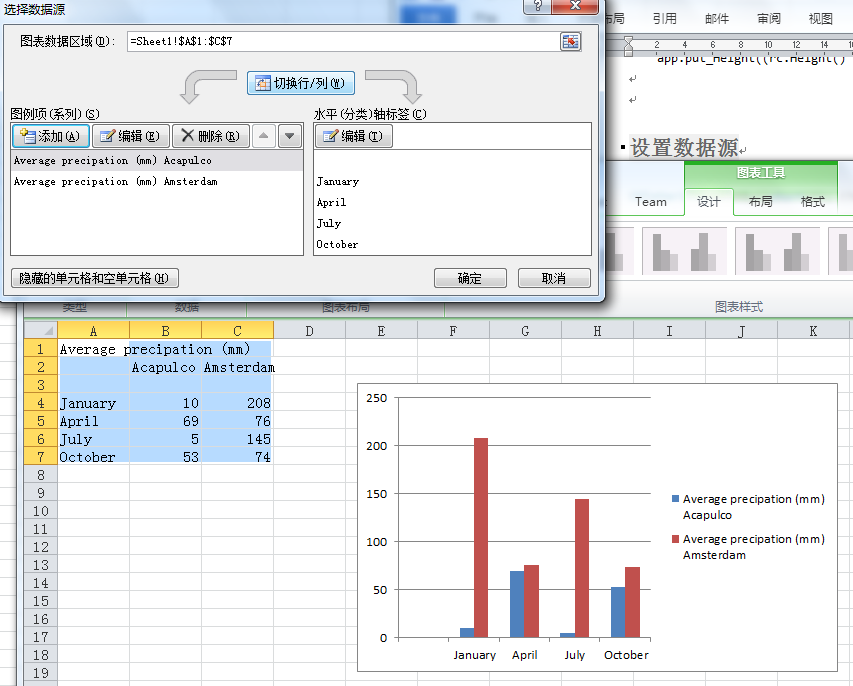
app.put\_Width(rc.Width() \* 3 / 4);

app.put\_Height((rc.Height() - 15) \* 3 / 4);

### 设置数据源

COleVariant plotBy((short)2); //xiColumns = 2 ：行是数据

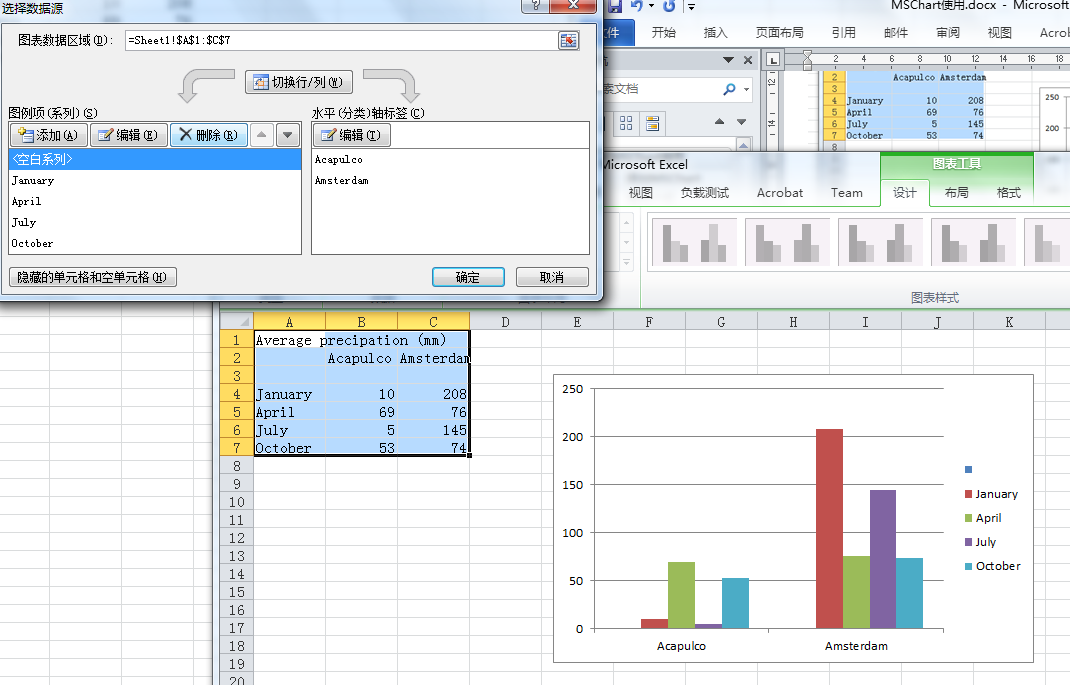
chart.SetSourceData(range, plotBy);



上面是

xiColumns = 2 的数据，说明January列作为x轴（分类），Acapulco作为系列

xiColumns = 1，Acapulco作为x轴，January作为系列，可能有空行，可以删除；



# Excel嵌入

## ActiveX

### 包容器

excel是一个自动化对象（ole对象）， 可作为ActiveX嵌入“包容器”；

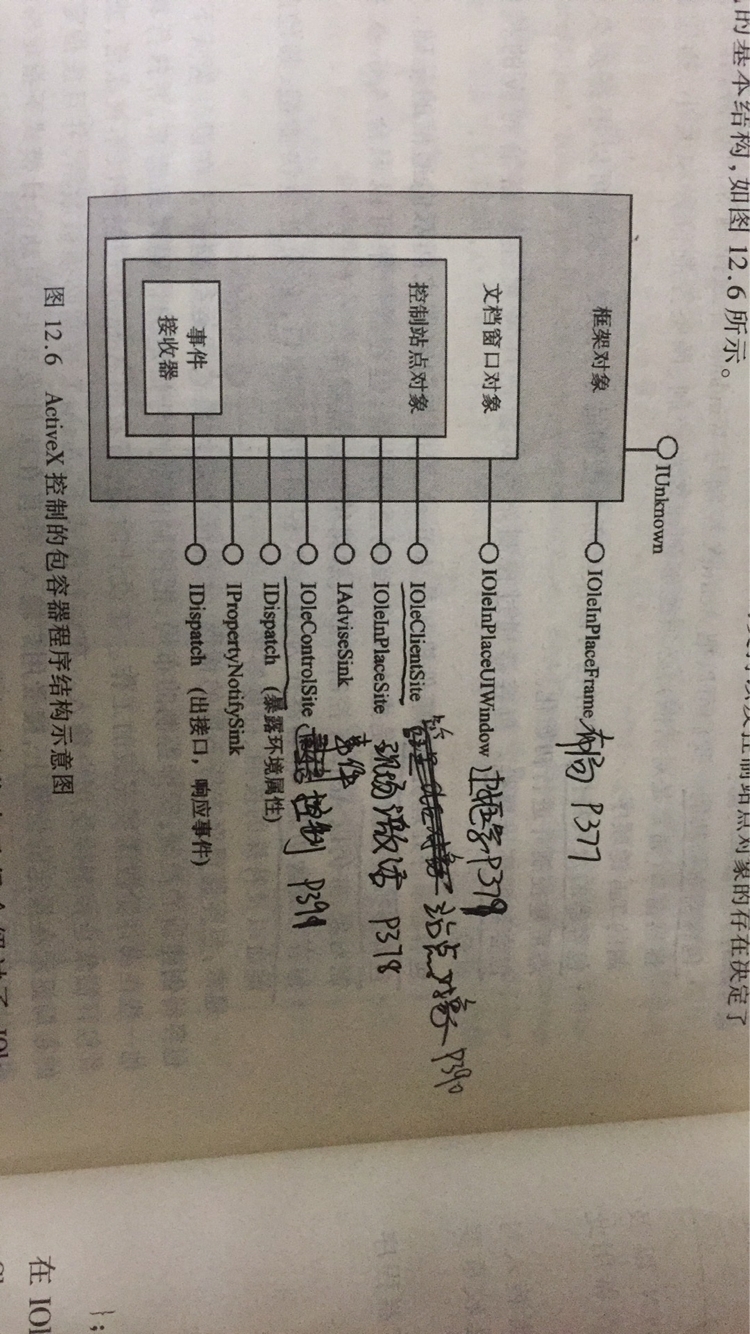
AppWizard勾选“Container”即可使程序变成包容器，dialog、doc/view。。。都可以是“包容器”；

“包容器”必须提供“站点对象”，每个“站点对象”是ActiveX和“包容器”交互的中介，“站点对象”对应mfc的**COleClientItem，**它实现了IOleClientSite、IAdviseSink、IOleInPlaceSite接口；

包容器还必须创建一个框架对象，实现IOleInPlaceFrame，ole通过它实现“菜单工具条等界面合并”；

包容器可以有多个文档窗口，每个窗口实现IOleInPlaceUIWindow；

“包容器”所使用的技术：**布局、永久、环境属性、事件、扩展控制、键盘**，



#### IAdivseSink

ole通过容器的IAdivseSink接口来接收**数据、视图**以及复合文档的变化消息，如图表数据更新、画面分割数变化。。。；

ole也可以通知容器它的各种变化，在ole的IDataObject::DAdvise中传递容器的这个IAdivseSink参数，然后IAdivseSink::OnDataChange通知容器；

这就是容器的“事件机制”，**COleClientItem**实现了它；

#### IOleClientSite

**COleClientItem**实现，ole通过它与容器通信；

#### IOlePlaceSite

包容器支持现场激活的标识；

管理激活，将ole菜单传递给容器。。。；

### 嵌入对象

对于嵌入对象（excel）要实现的接口，excel肯定都实现了，我们只需要了解怎么用即可；

IOleObject

《原理与应用，P390》，《internal+，P432》；

DoVerb 指示嵌入对象执行相应的动作，如OLEIVERB\_SHOW；

Close 、SetExtent。。。

原本自定义的ole是通过COleServerItem实现IOleObject的，而excel肯定是实现了的；

IPersistStorage

IDataObject

将变化通知容器，见**IAdivseSink**笔记；

IOleInPlaceObject

ole支持现场激活的标识；

IOleInPlaceActiveObject

ole处于激活状态时，

IOleObject 和 IDataObject实现在**COleServerItem**里，其他实现在COleServerDoc里，接口的作用在《internals+》中有；

## ActiveX标准分发ID

《原理与应用，P395》；

## MFC的支持

### 基本思想

mfc通过改进“文档/视图”结构来实现OLE文档；

基本思想：

**COleDocument** : CDocument

list<CDocItem>

IStorage\* 保存本地数据及ole对象

《internals+，P426》 成员&方法

CDocItem

**COleClientItem 站点对象**

内部有IOleClientSite、IAdviseSink、IOleWIndow、IOleInPlaceSite等接口；

**COleServerItem ole对象**

### 包容器

**COleClientItem** **+ COleDocument**

class CEmbed\_ExcelCntrItem : public COleClientItem

非常关键的一段代码：

CLSID clsid;

if (FAILED(::CLSIDFromProgID(L"Excel.sheet",&clsid)))

AfxThrowMemoryException();

// 关联“站点对象”与“ole对象”

if (!pItem->CreateNewItem(clsid))

而后边调用的：pItem->DoVerb(OLEIVERB\_SHOW, this);

实际是调用的IOleObject 的DoVerb，因为我们的excel肯定实现了IOleObject ；

IOleControl，IOleControlSite扩展ActiveX与站点对象的通信能力，如：键盘、扩展控制、环境属性、事件、属性页；

### ole服务器

**COleServerItem + COleServerDoc**

excel已经实现，但要了解接口，见《笔记：嵌入对象》；

### 容器/服务器的协调工作

OnFileNew-》CDocment::OnNewDocument-》

如果AppWizard选择了ole文档服务器-》EnableCompoundFile-》COleDocument::OnNewDocument-》m\_lpRootStorage；

用户插入ole对象-》COleDocument::list<CDocItem> ：“Insert New Object菜单”；

CreateNewItem实现 P437

创建substorage对象，名字：EmbeddingX-》GetItemStorage-》