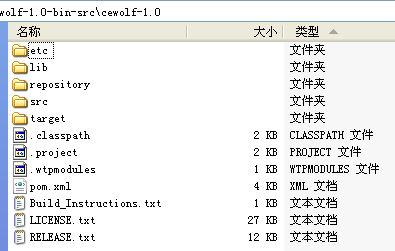
# Cewolf学习

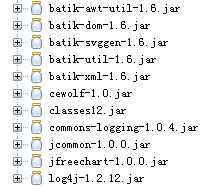
## Cewolf入门和环境的搭建

1. **Cwwolf简介：**
2. Cewolf是基于JFreechart的,利用JFreechart的绘制引擎,将最终的图片写入客户端.在服务器端不会创建任何文件.一切都是基于轻量级的会话对象和动态的数据分析的.
3. 写在JSP页面中的Cewolf,能用于基于web工程的Servlet/JSP,来制作复杂的统计图表.它提供几乎图的所有特征,象颜色,线条,图形等等.所以能够在不写任何JAVA代码的情况下让JSP页面显示图表.
4. Cewolf有一个单例的servlet,这个servlet用来处理图的绘画和标签库时,用这个标签库将在JSP页面的图的定义转成HTML img tag,servlet按HTML img tag绘制想要的图片送给客户端.
5. 它是开源的，运行在能编译servlet和servlet的容器中，图的类型非常丰富。
6. **环境的搭建**
7. 下载我们需要的jar。下载地址<http://cewolf.sourceforge.net/new/index.html,这里下载的是cewolf-1.0>

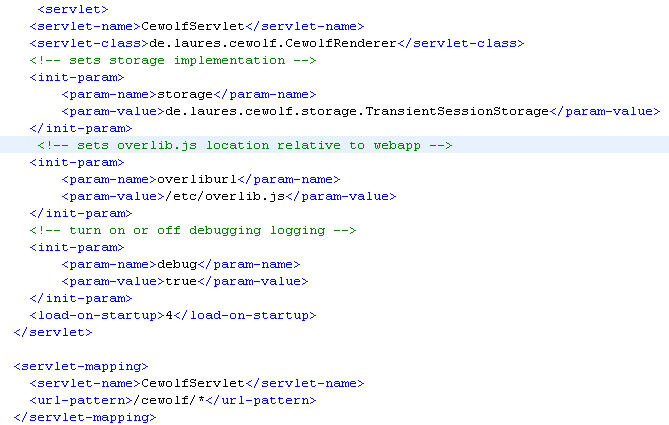
结构如图所示：



1. 将jar包下/Lib文件内容放在工程的/WEB-INF/lib 下面，如下图：(注意：Mozilla浏览器需要/example目录下的overlib.js，放入webapps根目录下)

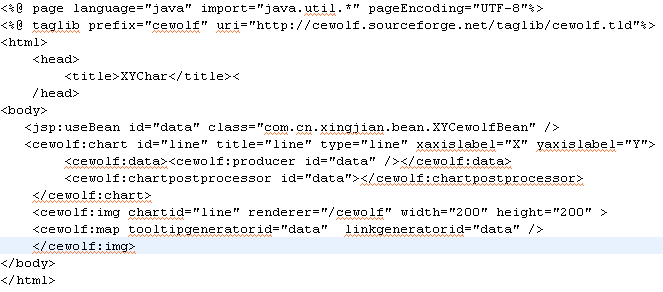


1. 将cewolf.tld放到WEB-INF下面,当引用标签的时候，我们加载这个文件就ok.(在etc目录下面)。
2. 在web.xml文件下配置servlet，如图：



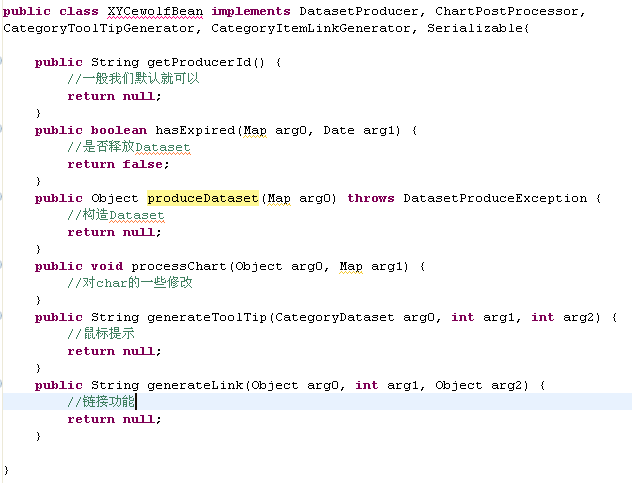
## 入门程序(简单例子)

1. 只是简单的例子.
2. 首先我们创建一个页面,页面很简单,引入cewolf标签就可以了,如下图:

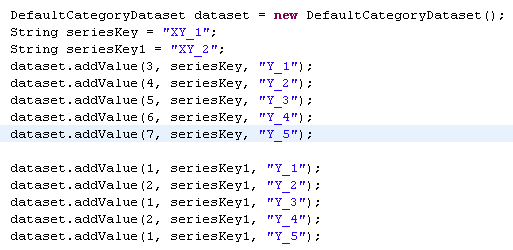


看到上面的jsp文件之后,我们就知道我们要写一个javabean,名字叫XYCewolfBean,先介绍一下页面里面的标签都是什么意思.

1. Title是char的标题,xaxislabel是x轴显示的内容,yaxislabel是y轴显示的内容。Id是char的标识,type是值char的类型。
2. <Cewolf:data>指的是数据其中包括<cewolf:producer id=”data” />这里面的这个id的值就是我们上面写的那个javabean的对象的名字，这里我们声明它的id=“data”.
3. <cewolf:chartpostprocessor id="data"></cewolf:chartpostprocessor>这个标签就是我们对默认生成的char不是很满意,需要我们对char进行一些修改来用的,同时也是要求我们在javabean的里面实现ChartPostProcessor，实现ChartPostProcessor方法，我们只要在这个方法里面来修改就可以了。
4. <cewolf:map tooltipgeneratorid="data" linkgeneratorid="data" />前面的tooltipgeneratorid是鼠标放上去提示的一些信息,后者是鼠标点击上面的时候的链接。也是和上面一样需要分别需要实现XXToolTipGenerator, XXLinkGenerator接口。这样就使得我们的图变得丰富一些。
5. <cewolf:img chartid="line" renderer="/cewolf" width="200" height="200" >这里面的renderer是我们在web文件里面配置好的,servlet。
6. 写好我的XYCewolfBean这个类。



方法的意思，里面已经标清楚了。在构造Dataset的方法里面我们写入这些：



这样我们的简单例子就完成了。请求页面就可以看见效果了。

## Cewolf的DataSet介绍

一个DatasetProducer需要实现3个方法，最重要的一个是produceDataset()方法，它实际上生成了一个用于画图的数据。这个方法用一个map作参数，这个map由许多指定的JSP标签填充，这个一会再介绍。

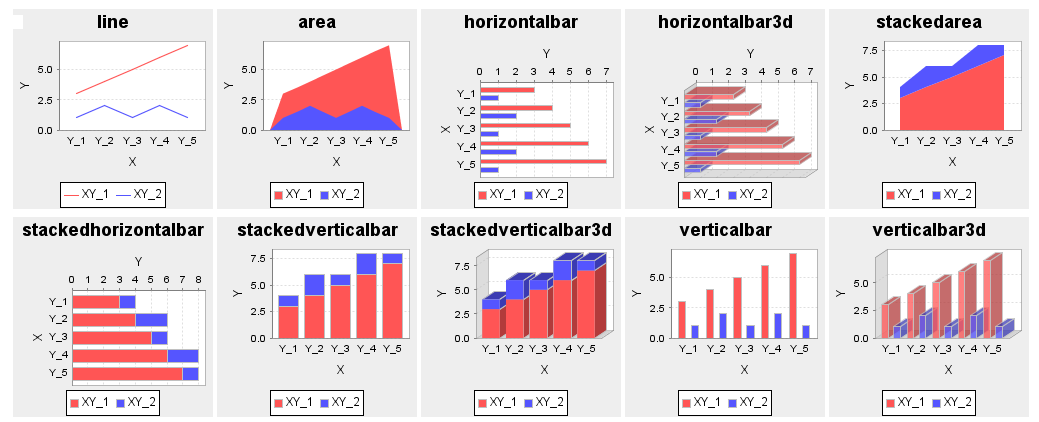
在当前生数据成器确实生成数据对象后，方法hasExpired()被才被Cewolf框架调用。当返回TRUE时，产生器将释放以前的数据。

Cewolf框架通过 getProducerId()方法提生成的一个唯一ID来标识一个生成者。2个具有相同ID生成实例将被认为时生成相同的数据。下面是几种常用的DataSet。

1. DefaultCategoryDataset(jfreechar)

它对应的char类型：area, horizontalbar,horizontalbar3d,line,stackedarea, stackedhorizontalbar,stackedverticalbar, stackedverticalbar3d, verticalbar

, verticalbar3d.对应如下：



1. XYDataSet(jfreechar)它对应的char类型：timeseries, scatter , xy , areaxy
2. PieDataset它对应char的类型：pie，pie3D
3. ValueDataset对应的char类型：meter
4. WindDataset对应的char类型：wind
5. OHLCDataset对应的char类型：highlow(DefaultHighlowDataSet), candlestick
6. IntervalCategoryDatase对应的char类型：gantt
7. tooltips下有4个接口：CategoryToolTipGenerator , PieToolTipGenerator

ToolTipGenerator ,XYToolTipGenerator根据不同的DataSet来实现不同的接口。

以上都是接口，使用的时候可以查看jfreechar的API来定制自己的实现类。

org.jfree.chart.axis.XXXAxis：用于处理图表的两个轴：纵轴和横轴

org.jfree.chart.render.XXXRender：负责如何显示一个图表对象

org.jfree.chart.urls.XXXURLGenerator:用于生成Web图表中每个项目的鼠标点击链接

XXXXXToolTipGenerator:用于生成图象的帮助提示，不同类型图表对应不同类型的工具提示类

## StandardLegend类-Jfreechar类方法

1. JFreeChart类

void setAntiAlias(boolean flag) 字体模糊边界

void setBackgroundImage(Image image) 背景图片

void setBackgroundImageAlignment(int alignment) 背景图片对齐方式（参数常量在org.jfree.ui.Align类中定义）

void setBackgroundImageAlpha(float alpha) 背景图片透明度（0.0～1.0）

void setBackgroundPaint(Paint paint) 背景色

void setBorderPaint(Paint paint) 边界线条颜色

void setBorderStroke(Stroke stroke) 边界线条笔触

void setBorderVisible(boolean visible) 边界线条是否可见

二. TextTitle类

void setFont(Font font) 标题字体

void setPaint(Paint paint) 标题字体颜色

void setText(String text) 标题内容

三. StandardLegend(Legend)类

void setBackgroundPaint(Paint paint) 图示背景色

void setTitle(String title) 图示标题内容

void setTitleFont(Font font) 图示标题字体

void setBoundingBoxArcWidth(int arcWidth) 图示边界圆角宽

void setBoundingBoxArcHeight(int arcHeight) 图示边界圆角高

void setOutlinePaint(Paint paint) 图示边界线条颜色

void setOutlineStroke(Stroke stroke) 图示边界线条笔触

void setDisplaySeriesLines(boolean flag) 图示项是否显示横线（折线图有效）

void setDisplaySeriesShapes(boolean flag) 图示项是否显示形状（折线图有效）

void setItemFont(Font font) 图示项字体

void setItemPaint(Paint paint) 图示项字体颜色

void setAnchor(int anchor) 图示在图表中的显示位置（参数常量在Legend类中定义）

## Axis类-Jfreechar类方法

一. Axis类

void setVisible(boolean flag) 坐标轴是否可见

void setAxisLinePaint(Paint paint) 坐标轴线条颜色（3D轴无效）

void setAxisLineStroke(Stroke stroke) 坐标轴线条笔触（3D轴无效）

void setAxisLineVisible(boolean visible) 坐标轴线条是否可见（3D轴无效）

void setFixedDimension(double dimension) （用于复合表中对多坐标轴的设置）

void setLabel(String label) 坐标轴标题

void setLabelFont(Font font) 坐标轴标题字体

void setLabelPaint(Paint paint) 坐标轴标题颜色

void setLabelAngle(double angle)` 坐标轴标题旋转角度（纵坐标可以旋转）

void setTickLabelFont(Font font) 坐标轴标尺值字体

void setTickLabelPaint(Paint paint) 坐标轴标尺值颜色

void setTickLabelsVisible(boolean flag) 坐标轴标尺值是否显示

void setTickMarkPaint(Paint paint) 坐标轴标尺颜色

void setTickMarkStroke(Stroke stroke) 坐标轴标尺笔触

void setTickMarksVisible(boolean flag) 坐标轴标尺是否显示

二. ValueAxis(Axis)类

void setAutoRange(boolean auto) 自动设置数据轴数据范围

void setAutoRangeMinimumSize(double size) 自动设置数据轴数据范围时数据范围的最小跨度

void setAutoTickUnitSelection(boolean flag) 数据轴的数据标签是否自动确定（默认为true）

void setFixedAutoRange(double length) 数据轴固定数据范围（设置100的话就是显示MAXVALUE到MAXVALUE-100那段数据范围）

void setInverted(boolean flag) 数据轴是否反向（默认为false）

void setLowerMargin(double margin) 数据轴下（左）边距

void setUpperMargin(double margin) 数据轴上（右）边距

void setLowerBound(double min) 数据轴上的显示最小值

void setUpperBound(double max) 数据轴上的显示最大值

void setPositiveArrowVisible(boolean visible) 是否显示正向箭头（3D轴无效）

void setNegativeArrowVisible(boolean visible) 是否显示反向箭头（3D轴无效）

void setVerticalTickLabels(boolean flag) 数据轴数据标签是否旋转到垂直

void setStandardTickUnits(TickUnitSource source) 数据轴的数据标签（可以只显示整数标签，需要将AutoTickUnitSelection设false）

三. NumberAxis(ValueAxis)类

void setAutoRangeIncludesZero(boolean flag) 是否强制在自动选择的数据范围中包含0

void setAutoRangeStickyZero(boolean flag) 是否强制在整个数据轴中包含0，即使0不在数据范围中

void setNumberFormatOverride(NumberFormat formatter) 数据轴数据标签的显示格式

void setTickUnit(NumberTickUnit unit) 数据轴的数据标签（需要将AutoTickUnitSelection设false）

四. DateAxis(ValueAxis)类：

void setMaximumDate(Date maximumDate) 日期轴上的最小日期

void setMinimumDate(Date minimumDate) 日期轴上的最大日期

void setRange(Date lower,Date upper) 日期轴范围

void setDateFormatOverride(DateFormat formatter) 日期轴日期标签的显示格式

void setTickUnit(DateTickUnit unit) 日期轴的日期标签（需要将AutoTickUnitSelection设false）

void setTickMarkPosition(DateTickMarkPosition position) 日期标签位置（参数常量在org.jfree.chart.axis.DateTickMarkPosition类中定义）

五. CategoryAxis(Axis)类

void setCategoryMargin(double margin) 分类轴边距

void setLowerMargin(double margin) 分类轴下（左）边距

void setUpperMargin(double margin) 分类轴上（右）边距

void setVerticalCategoryLabels(boolean flag) 分类轴标题是否旋转到垂直

void setMaxCategoryLabelWidthRatio(float ratio) 分类轴分类标签的最大宽度

## Plot类-Jfreechar类方法

一. Plot类

void setBackgroundImage(Image image) 数据区的背景图片

void setBackgroundImageAlignment(int alignment) 数据区的背景图片对齐方式（参数常量在org.jfree.ui.Align类中定义）

void setBackgroundPaint(Paint paint) 数据区的背景图片背景色

void setBackgroundAlpha(float alpha) 数据区的背景透明度（0.0～1.0）

void setForegroundAlpha(float alpha) 数据区的前景透明度（0.0～1.0）

void setDataAreaRatio(double ratio) 数据区占整个图表区的百分比

void setOutLinePaint(Paint paint) 数据区的边界线条颜色

void setOutLineStroke(Stroke stroke) 数据区的边界线条笔触

void setNoDataMessage(String message) 没有数据时显示的消息

void setNoDataMessageFont(Font font) 没有数据时显示的消息字体

void setNoDataMessagePaint(Paint paint) 没有数据时显示的消息颜色

二. CategoryPlot(Plot)类

void setDataset(CategoryDataset dataset) 数据区的2维数据表

void setColumnRenderingOrder(SortOrder order) 数据分类的排序方式

void setAxisOffset(Spacer offset) 坐标轴到数据区的间距

void setOrientation(PlotOrientation orientation) 数据区的方向（PlotOrientation.HORIZONTAL或PlotOrientation.VERTICAL）

void setDomainAxis(CategoryAxis axis) 数据区的分类轴

void setDomainAxisLocation(AxisLocation location) 分类轴的位置（参数常量在org.jfree.chart.axis.AxisLocation类中定义）

void setDomainGridlinesVisible(boolean visible) 分类轴网格是否可见

void setDomainGridlinePaint(Paint paint) 分类轴网格线条颜色

void setDomainGridlineStroke(Stroke stroke) 分类轴网格线条笔触

void setRangeAxis(ValueAxis axis) 数据区的数据轴

void setRangeAxisLocation(AxisLocation location) 数据轴的位置（参数常量在org.jfree.chart.axis.AxisLocation类中定义）

void setRangeGridlinesVisible(boolean visible) 数据轴网格是否可见

void setRangeGridlinePaint(Paint paint) 数据轴网格线条颜色

void setRangeGridlineStroke(Stroke stroke) 数据轴网格线条笔触

void setRenderer(CategoryItemRenderer renderer) 数据区的表示者（详见Renderer组）

void addAnnotation(CategoryAnnotation annotation) 给数据区加一个注释

void addRangeMarker(Marker marker,Layer layer) 给数据区加一个数值范围区域

三. PiePlot(Plot)类

void setDataset(PieDataset dataset) 数据区的1维数据表

void setIgnoreNullValues(boolean flag) 忽略无值的分类

void setCircular(boolean flag) 饼图是否一定是正圆

void setStartAngle(double angle) 饼图的初始角度

void setDirection(Rotation direction) 饼图的旋转方向

void setExplodePercent(int section,double percent) 抽取的那块（1维数据表的分类下标）以及抽取出来的距离（0.0～1.0），3D饼图无效

void setLabelBackgroundPaint(Paint paint) 分类标签的底色

void setLabelFont(Font font) 分类标签的字体

void setLabelPaint(Paint paint) 分类标签的字体颜色

void setLabelLinkMargin(double margin) 分类标签与图的连接线边距

void setLabelLinkPaint(Paint paint) 分类标签与图的连接线颜色

void setLabelLinkStroke(Stroke stroke) 分类标签与图的连接线笔触

void setLabelOutlinePaint(Paint paint) 分类标签边框颜色

void setLabelOutlineStroke(Paint paint) 分类标签边框笔触

void setLabelShadowPaint(Paint paint) 分类标签阴影颜色

void setMaximumLabelWidth(double width) 分类标签的最大长度（0.0～1.0）

void setPieIndex(int index) 饼图的索引（复合饼图中用到）

void setSectionOutlinePaint(int section,Paint paint) 指定分类饼的边框颜色

void setSectionOutlineStroke(int section,Stroke stroke) 指定分类饼的边框笔触

void setSectionPaint(int section,Paint paint) 指定分类饼的颜色

void setShadowPaint(Paint paint) 饼图的阴影颜色

void setShadowXOffset(double offset) 饼图的阴影相对图的水平偏移

void setShadowYOffset(double offset) 饼图的阴影相对图的垂直偏移

void setLabelGenerator(PieSectionLabelGenerator generator) 分类标签的格式，设置成null则整个标签包括连接线都不显示

void setToolTipGenerator(PieToolTipGenerator generator) MAP中鼠标移上的显示格式

void setURLGenerator(PieURLGenerator generator) MAP中钻取链接格式

四. PiePlot3D(PiePlot)类

void setDepthFactor(double factor) 3D饼图的Z轴高度（0.0～1.0）

五. MultiplePiePlot(Plot)类：

void setLimit(double limit) 每个饼图之间的数据关联（详细比较复杂）

void setPieChart(JFreeChart pieChart) 每个饼图的显示方式（见JFreeChart类个PiePlot类）

## Renderer类-Jfreechar类方法

一. AbstractRenderer类

void setItemLabelAnchorOffset(double offset) 数据标签的与数据点的偏移

void setItemLabelsVisible(boolean visible) 数据标签是否可见

void setItemLabelFont(Font font) 数据标签的字体

void setItemLabelPaint(Paint paint) 数据标签的字体颜色

void setItemLabelPosition(ItemLabelPosition position) 数据标签位置

void setPositiveItemLabelPosition(ItemLabelPosition position) 正数标签位置

void setNegativeItemLabelPosition(ItemLabelPosition position) 负数标签位置

void setOutLinePaint(Paint paint) 图形边框的线条颜色

void setOutLineStroke(Stroke stroke) 图形边框的线条笔触

void setPaint(Paint paint) 所有分类图形的颜色

void setShape(Shape shape) 所有分类图形的形状（如折线图的点）

void setStroke(Stroke stroke) 所有分类图形的笔触（如折线图的线）

void setSeriesItemLabelsVisible(int series,boolean visible) 指定分类的数据标签是否可见

void setSeriesItemLabelFont(int series,Font font) 指定分类的数据标签的字体

void setSeriesItemLabelPaint(int series,Paint paint) 指定分类的数据标签的字体颜色

void setSeriesItemLabelPosition(int series,ItemLabelPosition position) 数据标签位置

void setSeriesPositiveItemLabelPosition(int series,ItemLabelPosition position) 正数标签位置

void setSeriesNegativeItemLabelPosition(int series,ItemLabelPosition position) 负数标签位置

void setSeriesOutLinePaint(int series,Paint paint) 指定分类的图形边框的线条颜色

void setSeriesOutLineStroke(int series,Stroke stroke) 指定分类的图形边框的线条笔触

void setSeriesPaint(int series,Paint paint) 指定分类图形的颜色

void setSeriesShape(int series,Shape shape) 指定分类图形的形状（如折线图的点）

void setSeriesStroke(int series,Stroke stroke) 指定分类图形的笔触（如折线图的线

）

二. AbstractCategoryItemRenderer(AbstractRenderer)类

void setLabelGenerator(CategoryLabelGenerator generator) 数据标签的格式

void setToolTipGenerator(CategoryToolTipGenerator generator) MAP中鼠标移上的显示格式

void setItemURLGenerator(CategoryURLGenerator generator) MAP中钻取链接格式

void setSeriesLabelGenerator(int series,CategoryLabelGenerator generator) 指定分类的数据标签的格式

void setSeriesToolTipGenerator(int series,CategoryToolTipGenerator generator) 指定分

类的MAP中鼠标移上的显示格式

void setSeriesItemURLGenerator(int series,CategoryURLGenerator generator) 指定分类的

MAP中钻取链接格式

三. BarRenderer(AbstractCategoryItemRenderer)类

void setDrawBarOutline(boolean draw) 是否画图形边框

void setItemMargin(double percent) 每个BAR之间的间隔

void setMaxBarWidth(double percent) 每个BAR的最大宽度

void setMinimumBarLength(double min) 最短的BAR长度，避免数值太小而显示不出

void setPositiveItemLabelPositionFallback(ItemLabelPosition position) 无法在BAR中

显示的正数标签位置

void setNegativeItemLabelPositionFallback(ItemLabelPosition position) 无法在BAR中

显示的负数标签位置

四. BarRenderer3D(BarRenderer)类

void setWallPaint(Paint paint) 3D坐标轴的墙体颜色

五. LayeredBarRenderer(BarRenderer)类：

void setSeriesBarWidth(int series,double width) 设定每个分类的宽度（注意设置不要使

某分类被覆盖）

六. WaterfallBarRenderer(BarRenderer)类：

void setFirstBarPaint(Paint paint) 第一个柱图的颜色

void setLastBarPaint(Paint paint) 最后一个柱图的颜色

void setPositiveBarPaint(Paint paint) 正值柱图的颜色

void setNegativeBarPaint(Paint paint) 负值柱图的颜色

七. IntervalBarRenderer(BarRenderer)类：

需要传IntervalCategoryDataset作为数据源

八. GanttBarRenderer(IntervalBarRenderer)类：

void setCompletePaint(Paint paint) 完成进度颜色

void setIncompletePaint(Paint paint) 未完成进度颜色

void setStartPercent(double percent) 设置进度条在整条中的起始位置（0.0～1.0）

void setEndPercent(double percent) 设置进度条在整条中的结束位置（0.0～1.0）

九. StatisticBarRenderer(BarRenderer)类：

需要传StatisticCategoryDataset作为数据源

十. LineAndShapeRenderer(AbstractCategoryItemRenderer)类：

void setDrawLines(boolean draw) 是否折线的数据点之间用线连

void setDrawShapes(boolean draw) 是否折线的数据点根据分类使用不同的形状

void setShapesFilled(boolean filled) 所有分类是否填充数据点图形

void setSeriesShapesFilled(int series,boolean filled) 指定分类是否填充数据点图形

void setUseFillPaintForShapeOutline(boolean use) 指定是否填充数据点的Paint也被用于画数据点形状的边框

十一. LevelRenderer(AbstractCategoryItemRenderer)类：

void setItemMargin(double percent) 每个分类之间的间隔

void setMaxItemWidth(double percent) 每个分类的最大宽度

十二. CategoryStepRenderer(AbstractCategoryItemRenderer)类：

void setStagger(boolean shouldStagger) 不同分类的图是否交错

十三. MinMaxCategoryRenderer(AbstractCategoryItemRenderer)类：

void setDrawLines(boolean drawLines) 是否在每个分类线间画连接线

void setGroupPaint(Paint groupPaint) 一组图形连接线的颜色

void setGroupStroke(Stroke groupStroke) 一组图形连接线的笔触

void setMaxIcon(Icon maxIcon) 最大值的ICON

void setMinIcon(Icon minIcon) 最小值的ICON

void setObjectIcon(Icon objectIcon) 所有值的ICON

十四. AreaRender(AbstractCategoryItemRenderer)类：

没有特殊的设置

十五. StackedAreaRender(AreaRender)类：

没有特殊的设置

## 例子-Jfreechar类方法

1. 简单的例子,看一下renderer的过程.

String sFont = "宋体";

chart.setBorderVisible(true);

chart.setBorderPaint(new Color(0xFF,0x66,0x00));

chart.setBackgroundPaint(new Color(0xFF,0xF3,0xDE));

chart.getTitle().setPaint(Color.red);

chart.getTitle().setFont(new Font(sFont,Font.BOLD,14));

//设置Plot，不显示所有网格

((CategoryPlot)chart.getPlot()).setOutlinePaint(null);

((CategoryPlot)chart.getPlot()).setDomainGridlinesVisible(false);

((CategoryPlot)chart.getPlot()).setRangeGridlinesVisible(false);

//设置横轴字体，设定横轴轴线不可见，隐藏纵轴

((CategoryPlot)chart.getPlot()).getDomainAxis().setTickLabelFont(new Font(sFont,Font.PLAIN,12));

((CategoryPlot)chart.getPlot()).getDomainAxis().setAxisLineVisible(false);

((CategoryPlot)chart.getPlot()).getRangeAxis().setVisible(false);

//采用BarRenderer作为表示器

BarRenderer renderer = new BarRenderer();

renderer.setPaint(new GradientPaint(0.0f,0.0f,Color.orange,0.0f,0.0f,Color.yellow));

renderer.setOutlinePaint(Color.orange);

renderer.setDrawBarOutline(true);

//在条中央显示投票数值

renderer.setItemLabelAnchorOffset(-20.0f);

renderer.setLabelGenerator(new StandardCategoryLabelGenerator("{2}",new

DecimalFormat()));

renderer.setPositiveItemLabelPosition(new ItemLabelPosition());

renderer.setItemLabelsVisible(true)

CategoryPlot categoryplot = (CategoryPlot) jfreechart.getPlot();

CategoryItemRenderer categoryitemrenderer = categoryplot.getRenderer();

BarRenderer barrenderer = (BarRenderer) categoryitemrenderer;

// 拿到之后，额可以用Index直接设置

barrenderer.setSeriesPaint(3, new Color(255, 0, 0));

// 也可以根据系列的数值，获得Index，然后在设置颜色

barrenderer.setSeriesPaint(categorydataset.getRowIndex("<200米"), new Color(255, 0,

0));

下面我们写几个例子给大家看一下如下图：

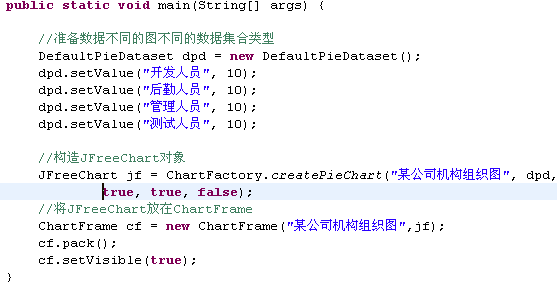
首先我们要登录<Http://www.jfree.org>去下载,jfreechar需要jcommon支持，所以我们将两个jar包都要下载下来。如下：



最好下载的时候也要下载源代码,如过没有文档的话，装一个ant(配置ANT\_HOME,JAVA\_HOME就可以了)进入ant文件，执行build.xml，但不是全执行，执行ant javadoc(build.xml里面的target)文档就生成了

下面的截图是一个简单的饼图,创建一个图一般就是怎么几个步骤：

图的组成一般包括：title，plot(中间的)，在plot下面在可以等到X,Y等等很多的，我都可以对其经行设置，还有一个就是lenged（下面提示的那块）创建的时候jfreechar的时候有个参数boolen就是设置是否显示这个lenged。



下面做一个柱状图



## 开发过程中遇到问题解决方式

1. Highlow图显示的是最高点，最低点，开盘值，收盘值，那么我们这样使用呢？

首先构造DefaultHighLowDataset



里面的参数第一个是曲线的名字，date是日期的数组，其余的都是double数组。

1. 绘制平均值的折线？

XYDataset dataset2 = MovingAverage.createMovingAverage(plot.getDataset(), "", 1 \* 24 \* 60 \* 60 \* 1000L, 0L);这是将DefaultHighLowDataset给这个函数来用算好平均值，返回一个XYDataset，里面可以调整算平均值的方式。

1. Polt上面添加线条怎么弄，该线条的美化改怎么弄？

plot.setDataset(1,DataSet),第一个参数是曲线的顺序，从0开始。

plot.setRenderer(1, render), ,第一个参数是曲线的顺序，从0开始。

1. Jfreechar1.0.0和1.0.13的版本差一个ohlc的包.使用的时候需要注意一下.有很大差异.
2. 设置曲线的颜色

用render的setPaint的方法就行.

1. 设置最大值和最小值(坐标轴)?

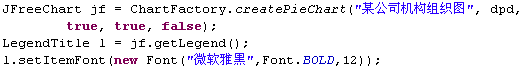
Y轴：plot.getRangeAxis().setupBound(数字); plot.getRangeAxis().setlowBound();

X轴：plot.getdomainAxis().setupbound(数字)plot.getDomainAxis().setLowBound();

1. 创建jfreechar，用什么？

我们一般用ChartFactory.createXXX();

1. 怎样修改Legend？



1. 怎样讲jfreechar对象转换成图片？(jfreechar只支持jpg，JPEG,png)



1. 坐标轴的字体倾斜？



1. 格式设置?

NumberAxis numberaxis = (NumberAxis)categoryplot.getRangeAxis();

 decimalFormat = new DecimalFormat("格式");          
         numberaxis.setNumberFormatOverride(decimalFormat);

1. 最小单位的控制

numberAxis.setStandardTickUnits(NumberAxis.createIntegerTickUnits()); // 设置Y轴为integer类型

1. 设置警戒线?

ValueMarker vm1 = new ValueMarker(valuesM[0]); vm1.setLabel(" 注意");vm1.setPaint(Color.orange);

plot.addRangeMarker(vm1);

1. 空心圈和圆点的?

XYLineAndShapeRenderer renderer = (XYLineAndShapeRenderer) plot.getRenderer();

renderer.setBaseShapesVisible(true);

//空心圈

renderer.setSeriesFillPaint(0, Color.red);

renderer.setSeriesFillPaint(1, Color.white);

renderer.setUseFillPaint(true);

//圆点

renderer.setDrawOutlines(true);

renderer.setSeriesOutlineStroke(0, new BasicStroke(2.0f));

renderer.setSeriesStroke(0, new BasicStroke(1.0f));

renderer.setSeriesShape(0, new Ellipse2D.Double(-2.5, -2.5, 5.0, 5.0));

1. 设置垂直?

DateAxis axis = (DateAxis) plot.getDomainAxis();

axis.setVerticalTickLabels(true);

1. 设置背景色和警戒区域？

TextTitle title = chart.getTitle();

title.setBorder(0, 0, 1, 0);

title.setBackgroundPaint(new GradientPaint(0f, 0f, Color.blue, 350f, 0f, Color.white, true));

title.setExpandToFitSpace(true);

chart.setBackgroundPaint(new GradientPaint(0f, 0f, Color.green, 350f, 0f, Color.white, true));

IntervalMarker target = new IntervalMarker(400.0, 600.0);

target.setLabel("Target Range");

target.setLabelFont(new Font("SansSerif", Font.ITALIC, 11));

target.setLabelAnchor(RectangleAnchor.LEFT);

target.setLabelTextAnchor(TextAnchor.CENTER\_LEFT);

target.setPaint(new Color(222, 222, 255, 128));

plot.addRangeMarker(target, Layer.BACKGROUND);

1. 设置鼠标提示?

**public** String generateToolTip(XYDataset arg0, **int** arg1, **int** arg2) {

String unit = "℃";

TimeSeriesCollection ds = (TimeSeriesCollection) arg0;

String y = String.*valueOf*(ds.getYValue(arg1, arg2));

Date dateTemp = **new** Date();

dateTemp.setTime((**long**)ds.getXValue(arg1, arg2));

String xDateString = "";

xDateString = sdf.format(dateTemp);

**return** xDateString+" "+y+" "+unit;

}