曲线模块开发文档

目录

[一、摘要 1](#_Toc111102168)

[项目引用 1](#_Toc111102169)

[第三方资源 1](#_Toc111102170)

[二、模块功能 1](#_Toc111102171)

[控件标识配置 1](#_Toc111102172)

[要求 2](#_Toc111102173)

[实现 2](#_Toc111102174)

[坐标轴 2](#_Toc111102175)

[要求 2](#_Toc111102176)

[实现 2](#_Toc111102177)

[曲线 2](#_Toc111102178)

[要求 3](#_Toc111102179)

[实现 3](#_Toc111102180)

[视图控制 3](#_Toc111102181)

[要求 3](#_Toc111102182)

[实现 4](#_Toc111102183)

[信息标识 4](#_Toc111102184)

[要求 4](#_Toc111102185)

[实现 4](#_Toc111102186)

# 一、摘要

该模块的主要功能为针对输入的坐标点数据，生成相应的曲线图形。

## 项目引用

该模块引用了公司内部其他软件项目的源代码。

* LoggingModule
  + 包装了 Serilog 并实现了日志功能。

## 第三方资源

曲线模块的实现过程中使用了一些第三方资源，所有资源的许可协议均为开源免费且不限制商用。

* [OxyPlot](https://github.com/oxyplot/oxyplot)
  + 支持多种曲线绘制模式的曲线图形绘制库。
  + 许可协议：MIT license
* [Prism](https://prismlibrary.com/)
  + 相当成熟的开发框架，旨在帮助用户更加容易的设计和构建丰富、灵活、易于维护 WPF 桌面应用程序。
  + 许可协议：MIT license
* [Serilog](https://github.com/serilog/serilog)
  + 支持多种功能的日志功能库，通过简单配置即可实现文件输出、模板配置和附加运行环境信息等功能
  + 许可协议：Apache-2.0 license

# 二、模块功能

## 控件标识配置

同一应用程序内可能同时存在多个曲线控件。每个控件必须被分配唯一的标识来确保控件的独立性。

### 要求

通过简便的方式为控件分配标识，同时针对 [Prism](https://prismlibrary.com/) 开发环境进行优化。

### 实现

为控件添加 ID 属性。该属性可直接在 XMAL 文件中定义，也可以通过 [Prism](https://prismlibrary.com/) 的 Region 导航功能来分配。

<PlotCtrlLib:PlotControl ID="MyPlot" />

regionManager.RequestNavigate(

RegionNames.MyRegion,

nameof(PlotControl),

new NavigationParameters() { { "ID", "MyPlot" } }

);

## 坐标轴

坐标轴功能是曲线绘制的前提。某些坐标轴属性也会影响曲线的显示效果。

### 要求

坐标轴的颜色、刻度与位置等属性均可根据实际要求进行自定义，从而实现 XY 轴和双 Y 轴的配置。

对于输入的不合理参数，模块需进行过滤处理，并将情况写入日志。

### 实现

开发者可以通过 PlotControlLibrary.Common.Configure 类中的相关方法创建和修改坐标轴。这种方式可以确保模块过滤掉不合理的参数。

var myAxis = Configure.CreateLinearAxis(position, title, color);

Configure.LinearAxis(myAxis, title: value);

配置完成的坐标轴将以 [Prism](https://prismlibrary.com/) 事件的方式发送至目标控件中。

eventAggregator  
 .GetEvent<AddAxisEvent>()  
 .Publish(new AddAxisEventPayload(targetList, axes));

## 曲线

试验数据在经过处理后可转化为坐标点序列的形式。坐标点序列在经过配置后传入曲线控件，最终成为曲线展示给用户。

### 要求

将坐标点数据转化为曲线，载入时间在 3 秒内，且平移、缩放等操作无明显卡顿。控件应当能一次性显示一条包含 2000000 个坐标点的曲线，或是一次性显示 200 条曲线，每条曲线包含 10000 个坐标点。

对于数据量大且需要实时展示一部分尾部内容的曲线，其刷新间隔需保持在 50ms 以下，从而保证曲线能流畅地进行动态展示。

曲线的样式、标题等属性可以自定义，以满足使用场景的需求。

对于输入的不合理参数，模块需进行过滤处理，并将情况写入日志。

### 实现

开发者可以通过 PlotControlLibrary.Common.Configure 类中的相关方法创建和修改曲线。这种方式可以确保模块过滤掉不合理的参数。

var myLineSeries = Configure.CreateLineSeries(  
 xAxisKey,  
 yAxisKey,  
 title,  
 color  
);  
Configure.LineSeries(myLineSeries, newXAxisKey, newYAxisKey);

配置完成的曲线坐标点序列将以 [Prism](https://prismlibrary.com/) 事件的方式发送至目标控件中。

eventAggregator  
 .GetEvent<AddSeriesEvent>()  
 .Publish(new AddSeriesEventPayload(targetList, series));

经测试，使用随机数据生成的包含 2000000 个数据点的曲线的所需的渲染时间大约 0.3 秒。200 条曲线(每条曲线10000个数据点)所需的渲染时间大约0.5秒。单条动态成长的曲线（可视区域内大约有2000个数据点）可以做到平滑无卡顿。

## 视图控制

视图控制包括对曲线和坐标轴的平移与缩放操作。

### 要求

视图缩放可通过鼠标滚轮、键盘的上下翻页键以及拖拽选框的方式实现。单步缩放倍率可自定义。

平移可通过键盘方向键和鼠标拖拽的方式实现。单步平移倍率可自定义。

视图可通过双击鼠标中键、键盘的 Home 键还原为默认配置。

### 实现

所有要求功能均已实现。对倍率的修改操作将以 [Prism](https://prismlibrary.com/) 事件的方式发送至目标控件中。当鼠标在坐标轴上滚动滚轮时，可以仅针对该坐标轴进行缩放。

eventAggregator  
 .GetEvent<ModifyPropsEvent>()  
 .Publish(  
 new ModifyPropsEventPayload(  
 targetList,  
 zoomFactor,  
 panFactor  
 )  
 );

另外鼠标滚轮、键盘的上下翻页键缩放以及键盘方向键平移功能都附带了精准模式。只要在按下 Ctrl 键的同时执行相关操作，那么平移和缩放倍率就会变为设定值的十分之一，从而让用户能更加精确地进行控制。

## 信息标识

附加于图表上的文字或图像内容，用于对曲线内容的指示说明。

### 要求

展示曲线时，可根据需要显示相关的曲线标注。通过特定操作可显示曲线上的具体坐标点的 XY 值。对于指定的特征点可进行高亮显示。

### 实现

曲线控件默认配置了标注显示功能。设置了 Title 属性的曲线会自动出现在标注中。

按住鼠标右键的同时将指针移到曲线上即可显示距离最近的坐标点数据(实现磁吸)。按住 Ctrl 键的同时按下鼠标右键则可以显示当前曲线位置的坐标数据（取消磁吸）。

特征点标注通过添加 PointAnnotation 实现。开发者可以通过 PlotControlLibrary.Common.Configure 类中的相关方法创建和修改 PointAnnotation。

myAnnotation = Configure.CreatePointAnnotation(

xAxisKey,

yAxisKey,

AnnotationX,

AnnotationY,

5,

color,

text,

shape

);

配置完成的 PointAnnotation 可通过 Prism 的事件机制发送给曲线控件。

eventAggregator

.GetEvent<AddAnnotationEvent>()

.Publish(

new AddAnnotationEventPayload(targetList, annotations)

);

用户可以在已添加的PointAnnotation上按住鼠标左键进行拖动，以此来调整特征点位置。