Syslab Online Python 绘图

服务端渲染方案设计



|  |  |  |  |
| --- | --- | --- | --- |
| 编制 | 胥博学 | 生效日期 | 2023-04-14 |
| 审核 |  | 批准 |  |

文件变更摘要

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 日期 | 版本 | 变更说明 | 修订 | 审核 | 批准 |
| 2023-04-14 | V0.1 | 初始建立 |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[1. 概述 3](#_Toc133414305)

[1.1 背景 3](#_Toc133414306)

[1.2 需求分析 3](#_Toc133414307)

[2. 总体架构 5](#_Toc133414308)

[3. 方案设计 5](#_Toc133414309)

[3.1 服务端方案设计 6](#_Toc133414310)

[3.1.1 WebAgg 6](#_Toc133414311)

[3.1.2 基于WebAgg的渲染流程 6](#_Toc133414312)

[3.2 通信方案设计 7](#_Toc133414313)

[3.2.1 通信生命周期 7](#_Toc133414314)

[3.2.2 通信模式 8](#_Toc133414315)

[3.2.3 WebSocket通信协议 9](#_Toc133414316)

[3.2.4 WebSocket协议的优势 9](#_Toc133414317)

[3.2.5 WebSocket协议的劣势 10](#_Toc133414318)

[3.3 异常处理机制 10](#_Toc133414319)

[3.4 绘图种类及交互种类 11](#_Toc133414320)

[3.4.1 绘图种类 11](#_Toc133414321)

[3.4.2 交互种类 11](#_Toc133414322)

[3.5 前端交互设计 11](#_Toc133414323)

[3.5.1 功能描述 11](#_Toc133414324)

[3.5.2 交互流程 12](#_Toc133414325)

[3.5.3 界面设计 12](#_Toc133414326)

[3.5.4 相关原理 15](#_Toc133414327)

[3.6 后端接口设计 16](#_Toc133414328)

[3.6.1 交互类事件接口 16](#_Toc133414329)

[3.6.2 修改属性接口 16](#_Toc133414330)

[3.6.3 鼠标交互事件接口协议 17](#_Toc133414331)

[3.6.4 更新工具栏状态接口协议 18](#_Toc133414332)

[3.6.5 更新右键菜单接口协议 18](#_Toc133414333)

[3.6.6 更新属性检查器接口协议 18](#_Toc133414334)

[3.6.7 设置属性接口协议 19](#_Toc133414335)

[4. 最终交付形态 20](#_Toc133414336)

[5. 风险点及拟解决方案 20](#_Toc133414337)

[5.1 Websocket服务关闭及figure实例的回收 20](#_Toc133414338)

[5.2 通信稳定性 20](#_Toc133414339)

[5.3 网络延迟 21](#_Toc133414340)

[5.4 同时支持python绘图、julia绘图、python调julia绘图 21](#_Toc133414341)

[6. 服务端渲染与客户端渲染方案对比 21](#_Toc133414342)

# 概述

## 背景

Syslab Online现阶段Python绘图功能不完善，需要支持调用Matplotlib库绘图，当前Syslab Online中Python脚本运行在服务端，实现Python绘图显示的核心就是将服务端的绘图信息传递到前端。

## 需求分析

1. 功能需求

表 1-1功能需求

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 指标类别 | 具体指标 | 优先级 | Demo |
| 功能 | 绘图 | 满足折线图、散点图、波特图、simth图、柱状图、天线方向图2d3d、热力图，并包含颜色、坐标轴属性； | ★★★ | ☑ |
| 可视化交互 | 支持旋转、平移、缩放等基础交互功能； | ★★ | ☑ |
| 支持选中曲线、散点并修改属性； | ★★ | ☑ |
| 支持修改坐标轴属性、双y轴、对数坐标轴 | ★★ | □ |
| 主次网格配置、随缩放动态改变 | ★★ | □ |
| 游标、tooltip | ★★ | □ |
| 支持属性面板查看/修改属性 | ★★ | □ |
| 导入导出 | 支持数据导入导出、绘图图片的导出； | ★★ | □ |

1. 性能要求

表 1-2性能要求

|  |  |  |  |
| --- | --- | --- | --- |
|  | 指标类别 | 具体指标 | 优先级 |
| 性能 | 绘图 | 常规硬件配置下支持千万级数据点绘制； | ★★★ |
| 可视区域数据尽可能不丢失且同时保证流畅度 | ★★★ |
| 导入导出 | 常规硬件配置下千万级数据导入耗时≤15s； | ★★★ |
| 常规硬件配置下千万级数据导出耗时≤15s； | ★★★ |
| 常规硬件配置下导出图片耗时≤5s； | ★★★ |
| 其他 | 高兼容性 | 支持Chrome、Edge、Firefox等主流浏览器； | ★★★ |
| 高扩展性 | 支持后续绘图种类以及交互方式的扩展； | ★★★ |

# 总体架构

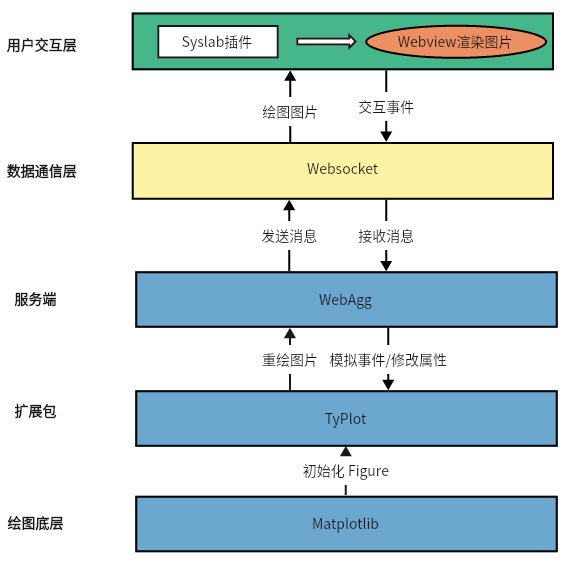


图 2-1总体架构

# 方案设计

本方案为基于Matplotlib WebAgg的服务端渲染绘图方案。以下为详细的方案设计。

## 服务端方案设计

### WebAgg

WebAgg 是由 Matplotlib 提供的扩展，用于在 Web 应用程序中创建和显示图形。它提供了一个 Web 服务器端渲染引擎，可以将 Matplotlib 的图表转换为 Web 应用程序中的 HTML 元素，以便在 Web 浏览器中显示。

使用 WebAgg，用户可以将 Matplotlib 图表嵌入到 Web 应用程序中，例如 Web 页面、交互式 Web 应用程序等。WebAgg 支持多种 Matplotlib 图表类型，包括折线图、散点图、柱状图、饼图等等。同时，WebAgg 也支持自定义图表样式和配色方案，以及支持远程数据源的交互式图表。

### 基于WebAgg的渲染流程

1. **服务端—Web前端：**
2. 创建 Matplotlib Figure。
3. 创建websocket服务。
4. 将Figure转为图片。
5. 使用 websocket通信将图片传到Web前端。
6. Web前端接收消息。
7. Web前端渲染图片。
8. **Web前端—服务端：**
9. 界面交互触发事件。
10. 事件管理分类
11. 通过websocket通信将事件传到服务端
12. 服务端接收消息
13. 根据事件种类调用Matplotlib的API修改Figure
14. 将修改后的Figure转为图片
15. 使用 websocket通信将图片传到Web前端更新视图。

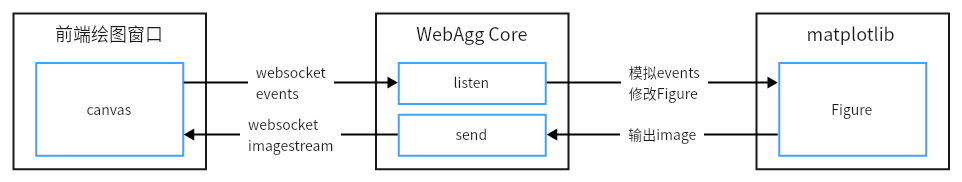


图 3-1服务端渲染流程

## 通信方案设计

### 通信生命周期

1. **绘图创建：**当WebAgg Core服务启动后，使用python绘图创建Figure对象，再使用TyPlot扩展库对原始Figure实例进行初始化（初始化状态及交互绑定），并将其挂载到WebAgg上。同时通知Syslab server创建绘图窗口，并使用figureId初始化websocket链接。

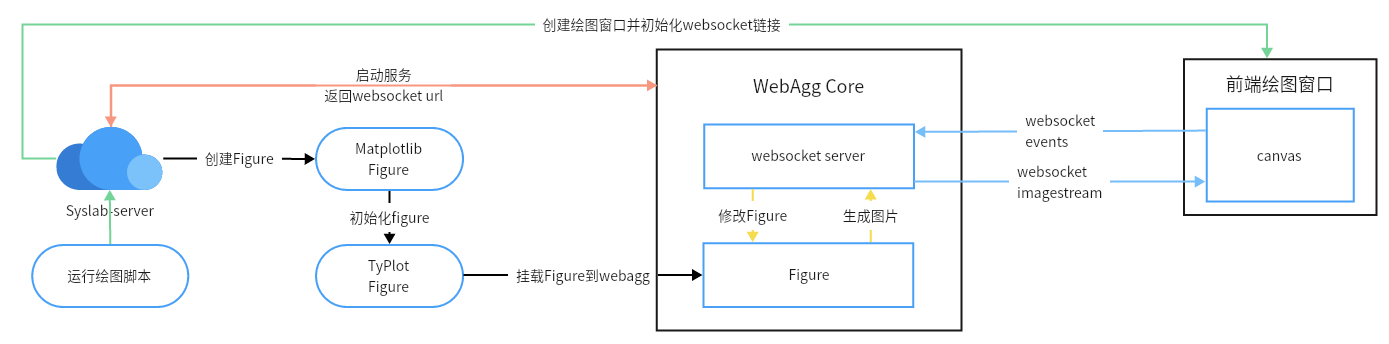


图 3.2.1-1 绘图初始化数据流图

1. **绘图交互：**Web应用程序在绘图窗口上添加鼠标事件监听器等交互功能，并将绘图窗口上的鼠标交互事件和属性检查器的修改事件通过websocket链接发送到WebAgg Core的websocket server。WebAgg Core接收到消息后会对其处理并进行事件分发，分配到相应的API接口。当Figure实例修改后，会重新生成绘图图片并通过websocket发给前端绘图窗口，前端绘图窗口接收消息后通过消息分发将图片通过canvas渲染到窗口上。当Figure实例的工具栏/右键菜单/属性面板改变，会把最新状态通过websocket发给前端绘图窗口，前端绘图窗口接收消息后通过消息分发将最新状态同步到工具栏/右键菜单/属性面板。

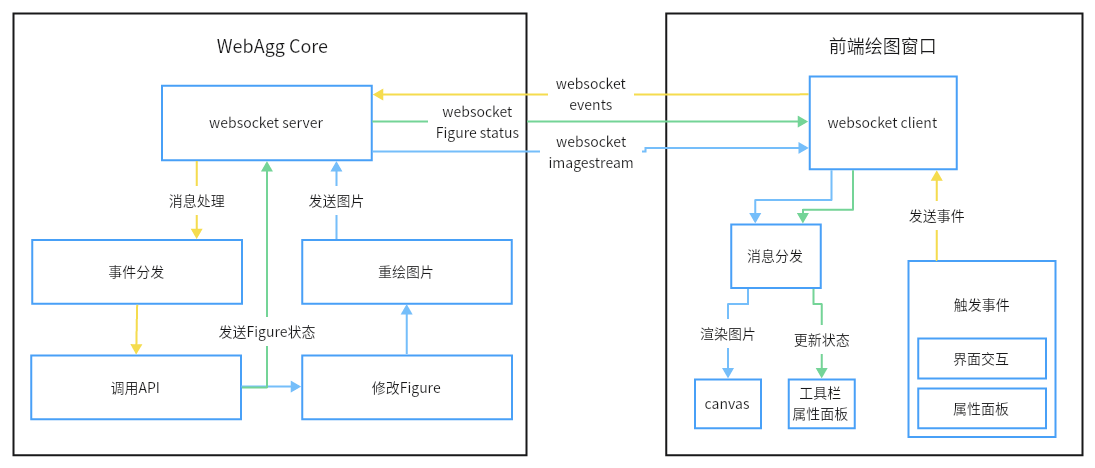


图 3.2.1-2 绘图交互数据流图

1. **绘图销毁：**当绘图窗口关闭时，Syslab Server将销毁绘图窗口并关闭其websocket链接。WebAgg Core监听到链接关闭时，将销毁其websocket链接并根据其figureId回收WebAgg上的Figure实例。

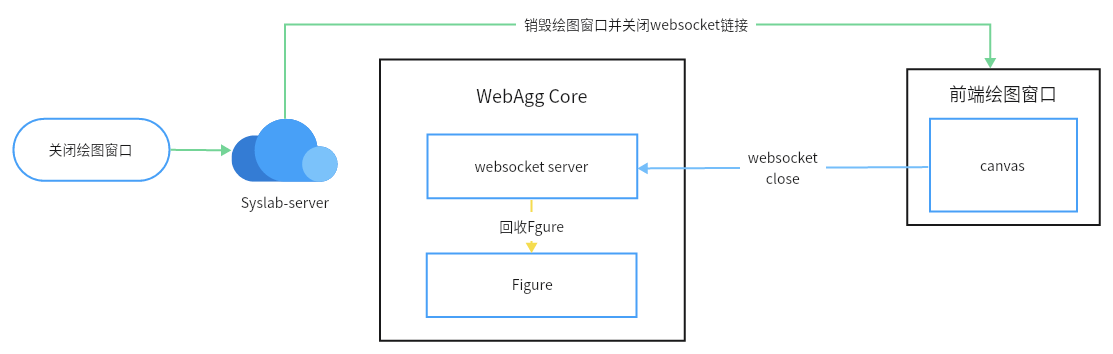


图 3.2.1-3 绘图销毁数据流图

### 通信模式

一个Figure实例与一个 绘图窗口对应。每个用户所有的Figure实例都挂载到一个WebAgg服务上，通过唯一figureId进行区分并生成对应的websocket url，每个通信都是相互独立互不干涉的。这样即避免了多个服务引起的性能开销，又保证了多个绘图的相互独立。

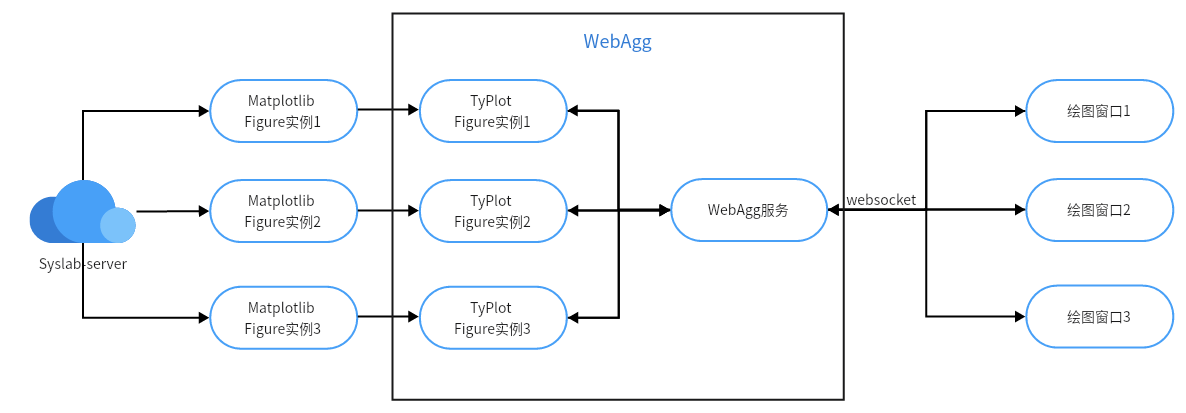


图 3.2.2-1 通信模式

### WebSocket通信协议

WebSocket是一种双向通信协议，用于在Web浏览器和Web服务器之间建立实时连接。WebSocket的主要特点包括：

1. **实时性：**WebSocket支持双向实时通信，数据可以在客户端和服务器之间实时传输，不需要进行HTTP请求和响应。
2. **长连接：**WebSocket连接会持久保持，当客户端和服务器之间建立websocket连接后，双方都可以随时发送和接收数据，直到其中一方主动关闭连接。
3. **二进制数据传输：**WebSocket可以传输二进制数据，这使得WebSocket比传M的HTTP请求和响应更加高效。

### WebSocket协议的优势

考虑到千万级数据绘图的情况下，传输数据量会很大，这时就需要选择传输效率更加高效的方案。

两者在传输数据量较大的情况下，显然是WebSocket通信协议更加高效，理由如下：

1. WebSocket支持二进制数据传输，这使得它可以更高效地传输大量的数据。相比之下，JSON-RPC主要使用JSON格式作为数据传输格式，虽然JSON格式在处理文本数据时非常高效，但是当需要传输大量的数据时，JSON格式的开销会变得更大。
2. WebSocket的长连接机制也使得它更适合传输大数据。因为WebSocket连接会持久保持，双方可以随时发送和接收数据，而不需要在每次数据传输时重新建立连接，这使得WebSocket在传输大数据时更加稳定和高效。

### WebSocket协议的劣势

1. 连接稳定性：相比于 HTTP 协议，WebSocket 连接更加不稳定，容易因为网络状况等因素导致连接中断。
2. 数据大小限制:WebSocket 协议没有像 HTTP 协议那样对数据大小进行限制，但是在实际应用中，由于网络传输损耗等因素，数据大小可能会受到一定限制。
3. 不支持缓存:WebSocket 通信不支持缓存，这意味着每次请求都需要重新发起，从而提高了请求次数和网络负载。

## 异常处理机制

1. 网络异常
2. websocket连接失败或者超时，可能原因是服务异常
3. websocket连接成功，客户端或者服务端有可能主动断开连接（调用close方法）
4. websocket连接正常，心跳出现异常
5. websocket连接正常，心跳正常，数据发送出现异常
6. websocket连接正常，心跳正常，数据接受出现异常
7. websocket连接正常，心跳正常，数据传输过大导致异常
8. websocket连接正常，心跳正常，数据传输过于频繁导致异常

## 绘图种类及交互种类

### 绘图种类

1. 线图

* Plot二维线图
* Semilogx
* Semilogy
* Loglog
* Area
* Errorbar
* Errorbarh
* Plot3

1. 针状图和阶梯图

* Stem
* Stairs
* Stem3

1. 条形图

* Bar
* Barh
* Bar3
* Bar3h
* Histogram
* Pareto
* Plotmatrix

1. 散点图

* Scatter
* Scatter3
* Spy

1. 饼图

* Pie

1. 极坐标图

* Polar
* Compass

1. 地理图

* Geoplot
* Geoscatter
* Geobubble

1. 等高线图

* Contour
* Contourf
* Contour3

1. 影像图

* Image
* Imagesc
* Imshow
* Heatmap

1. 三维曲面图

* Surf
* Surfc
* Mesh
* Meshc

1. 向量场图

* Feather
* Quiver
* Quiver3
* Streamline

### 交互种类

1. 鼠标操作

* 框选缩放、滚轮缩放
* 平移、旋转
* 选中

1. 线性选中及修改

* 选中线/点/面修改属性

1. 网格修改

* 正常网格（仅支持显隐、颜色、虚实线）
* 对数坐标网格
* 专业绘图网格
* 在有坐标轴的情况下主网格和次网格切换
* 坐标轴刻度跟随缩放变化
* 刻度自定义

1. 坐标轴修改

* 横坐标修改
* 纵坐标修改
* 自定义坐标轴
* 双y轴
* Label修改

1. 背景修改

* 背景颜色修改
* 背景透明度修改

1. 图例

* 图例的显隐

1. Colorbar

* 自定义颜色主题

1. 游标

* 游标拖动
* 数据对比

1. 数据标签

* 自定义数据标签
* 删除标签
* 添加标签

1. 右键菜单

* 选中线/点/面修改属性
* 不同的选中对象对应各自的右键菜单

1. 属性检查器

* 显示属性
* 修改属性

## 前端交互设计

### 功能描述

本功能针对Syslab Online绘图提供交互功能。

1. 缩放、平移、旋转、数据提示
2. 导出图片
3. 选中功能
4. 通过右键菜单及属性面板修改绘图属性
5. 通过Ribbon菜单修改绘图属性

### 交互流程

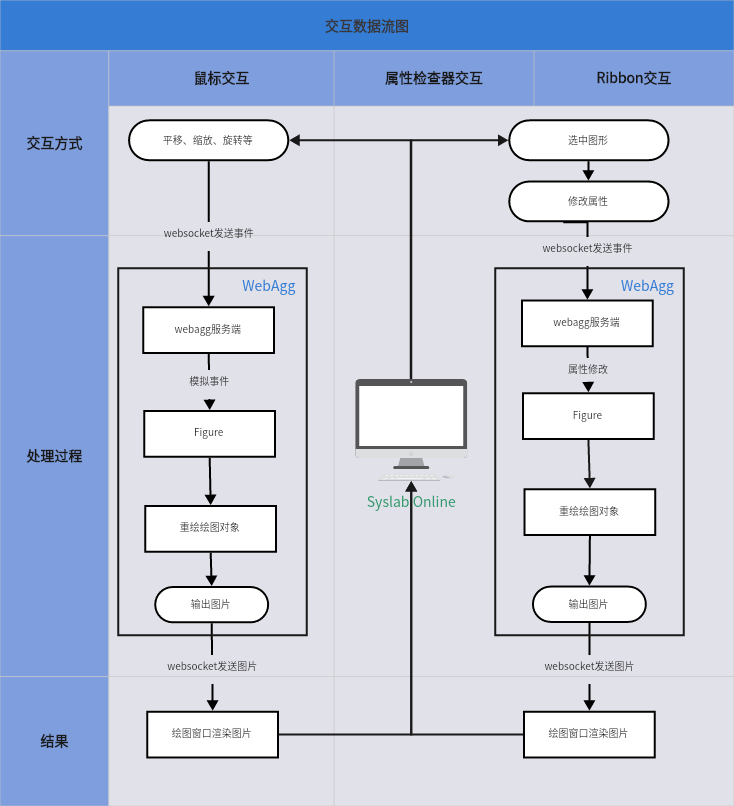


图 3.5.2-1 交互流程

### 界面设计

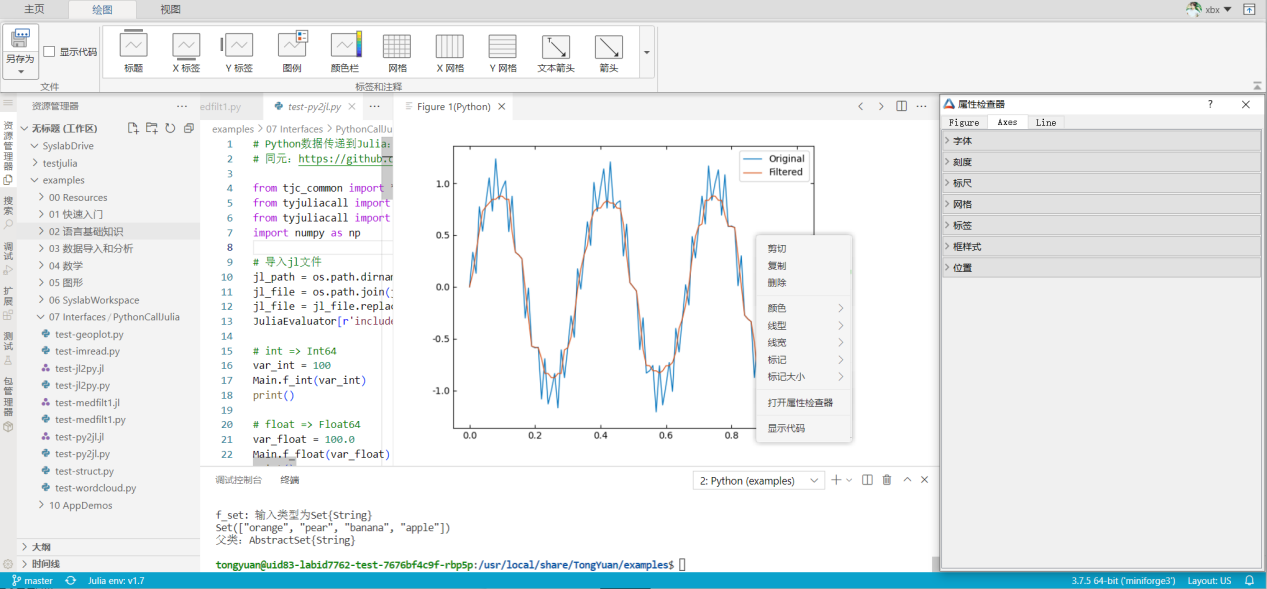


图 3‑5 整体界面设计

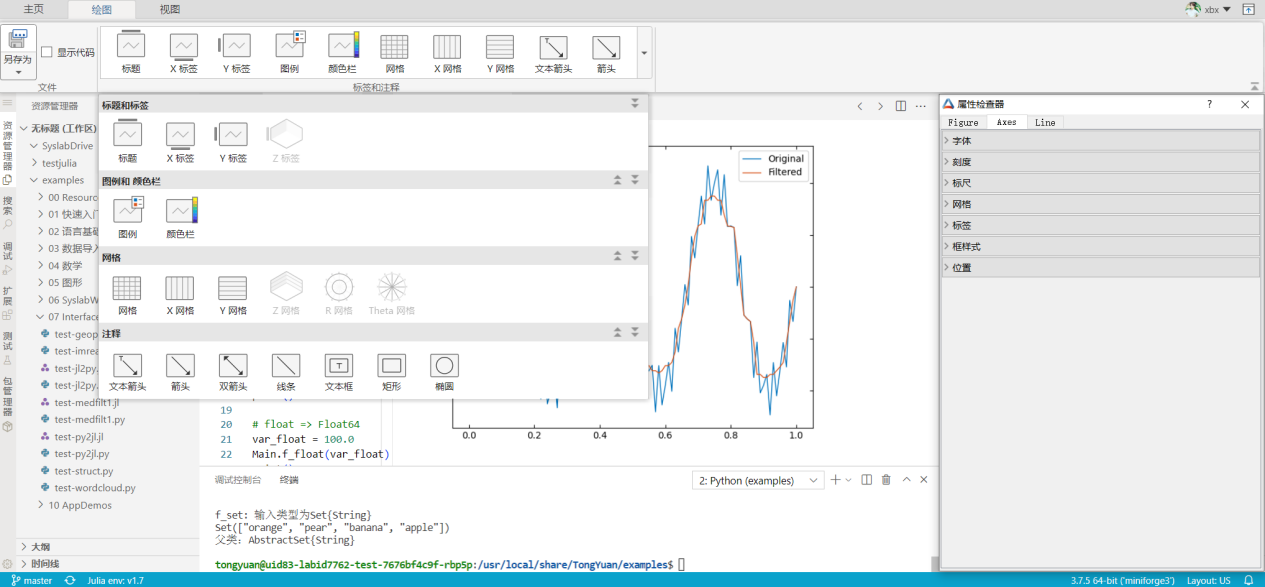


图 3‑6 Ribbon菜单

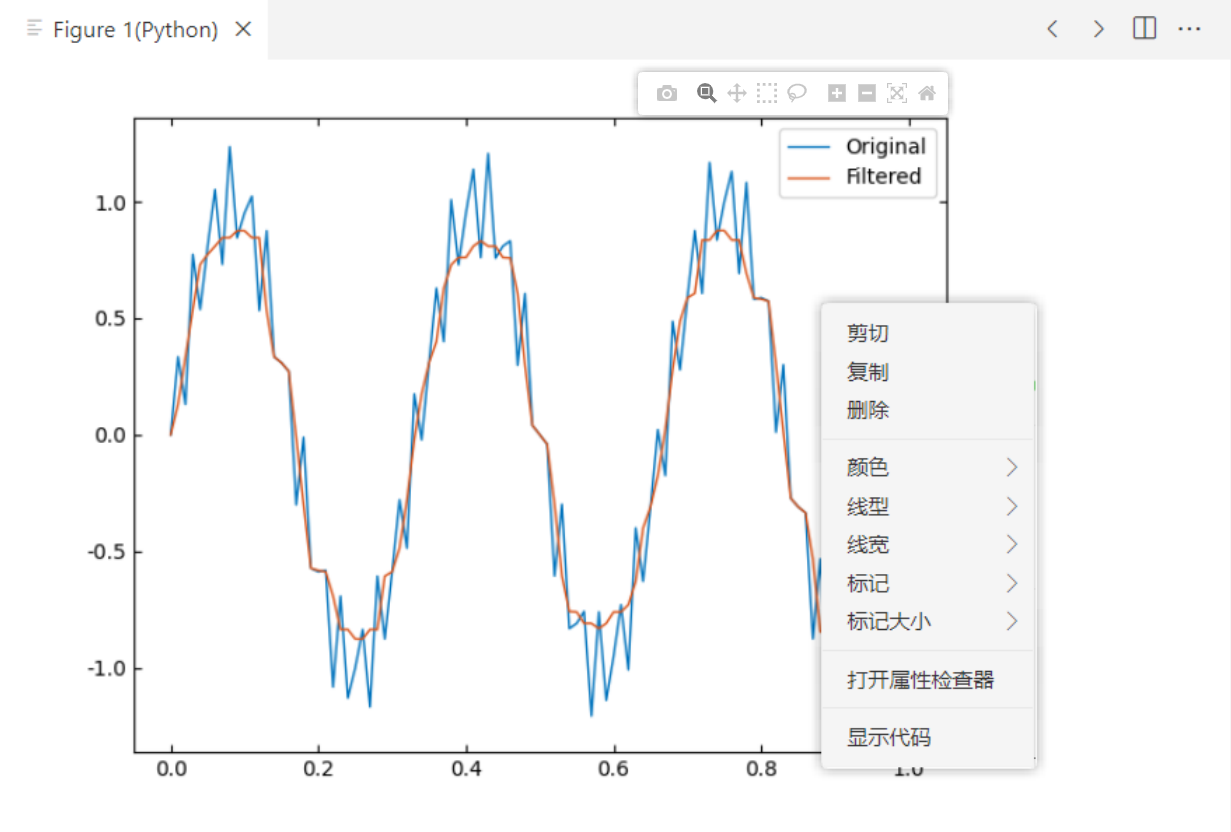


图 3‑7右键菜单

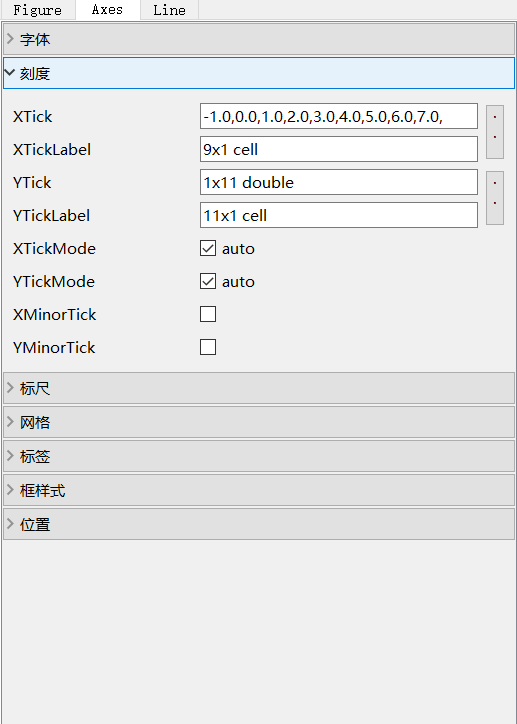


图 3‑8属性检查器

### 相关原理

1. 缩放、平移、旋转、数据提示：

通过鼠标事件计算出属性修改数值，调用更新API从而更新绘图图形。

1. 导出图片：

将当前绘图图形转为png/svg格式的图片，并保存到指定位置。

1. 选中功能：

通过鼠标事件获取选中的路径，并利用其id获取对应的绘图数据。

1. 通过右键菜单及属性面板修改绘图属性：

根据不同的激活目标生成对应的右键菜单内容，属性检查器包含该绘图对象所有属性，修改属性后通过通信发送到绘图窗口，接受消息后调用更新API从而更新绘图图形。

1. 通过Ribbon菜单修改绘图属性:

Ribbon菜单包含常用属性，其中的按钮会对当前激活的绘图窗口生效，修改属性后通过通信发送到绘图窗口，接受消息后调用更新API从而更新绘图图形。

## 后端接口设计

### 交互类事件接口

前端传输鼠标交互事件，后端根据交互类型进行相应的操作（例如选中点、线）后重绘图片。



图 3.6.1-1 交互类事件接口

### 修改属性接口

前端传输属性修改值，后端接收后针对激活的对象（例如线、点、坐标轴）进行相应属性的修改后重绘图片。



图 3.6.2-1 属性类交互接口

### 鼠标交互事件接口协议

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| type | String | True | 事件类型 | button\_press、button\_release、dblclick、motion\_notify、contextmenu  、wheel、figure\_enter、figure\_leave | 鼠标事件 |
| figure\_id | Int | False | Figure唯一id | 2254102997960 |  |
| x | Int | False | 鼠标x坐标 | 100 |  |
| y | Int |  | 鼠标y坐标 | 100 |  |
| button | Int |  | 鼠标按键标号 | 0/1/2 |  |
| step | Int |  | 缩放 | 1/-1 |  |
| guiEvent | Event |  | gui模拟事件 |  |  |

### 更新工具栏状态接口协议

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 参数 | |  | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| type |  |  | String | True | 事件类型 | update\_menu\_list | 更新右键菜单 |
| menu\_list |  |  | Array<Menu\_Item> | False | 右键菜单列表 |  |  |
|  | Menu\_Item |  | Object | True | 菜单项 |  |  |
|  |  | name | String | True | 菜单项名称 | ‘lineStyle’ |  |
|  |  | disabled | Boolean | False | 是否禁用 |  |  |

### 更新右键菜单接口协议

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 参数 | |  | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| type |  |  | String | True | 事件类型 | update\_toolbar\_list | 更新工具栏 |
| toolbar\_list |  |  | Array<Toolbar\_Item> | False | 右键菜单列表 |  |  |
|  | Toolbar\_Item |  | Object | True | 菜单项 |  |  |
|  |  | name | String | True | 菜单项名称 | ‘Pan’ |  |
|  |  | disabled | Boolean | False | 是否禁用 |  |  |
|  |  | visibility | Boolean | False | 是否显示 |  |  |

### 更新属性检查器接口协议（以曲线图为例）

#### 整体数据协议

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| type |  | Object | True | 颜色 | update\_figure\_attribute | 更新属性面板 |
| Figure |  | FIGURE | True | Figure对象 |  |  |
| Axes |  | AXES | True | 坐标轴 |  |  |
| Line |  | LINE | True | 线对象 |  |  |

#### FIGURE

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| Color |  | Object | True | 颜色 | update\_figure\_attribute | 更新属性面板 |
|  | BackgroundColor | String | True |  | plot |  |
| Setting |  | Object | True | 设置 |  |  |
|  | AutoAdjustView | Boolean | True |  |  |  |
|  | DataTipsOn | Boolean | True |  |  |  |

#### AXES

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| Font |  | Object | True | 字体 |  |  |
|  | FontName | String | True |  |  |  |
|  | FontAngle | Boolean | True |  |  |  |
|  | FontWeight | Boolean | True |  |  |  |
|  | FontSize | Number | True |  |  |  |
| Ticks |  | Object | True | 刻度 |  |  |
|  | XTick | Array<Number> | True |  |  |  |
|  | XTickLabel | Array<String> | True |  |  |  |
|  | YTick | Array<Number> | True |  |  |  |
|  | YTickLabel | Array<String> | True |  |  |  |
|  | XTickMode | Boolean | True |  |  |  |
|  | YTickMode | Boolean | True |  |  |  |
|  | XMinorTick | Boolean | True |  |  |  |
|  | YMinorTick | Boolean | True |  |  |  |
| Rulers |  |  |  | 标尺 |  |  |
|  | Xlim | Array<Number> | True |  |  |  |
|  | XlimMode | Boolean | True |  |  |  |
|  | Ylim | Array<Number> | True |  |  |  |
|  | YlimMode | Boolean | True |  |  |  |
|  | XColor | String | True |  |  |  |
|  | YColor | String | True |  |  |  |
|  | XScale | String | True |  |  |  |
|  | YScale | String | True |  |  |  |
| Grids |  | Object | True |  |  |  |
|  | XGrid | Boolean | True |  |  |  |
|  | YGrid | Boolean | True |  |  |  |
|  | GridLineStyle | String | True |  |  |  |
|  | GirdLineColor | String | True |  |  |  |
|  | GridLineAlpha | Number | True |  |  |  |
|  | XMinorGird | Boolean | True |  |  |  |
|  | YMinorGird | Boolean | True |  |  |  |
|  | Minor GridLineStyle | String | True |  |  |  |
|  | Minor GirdLineColor | String | True |  |  |  |
|  | Minor GridLineAlpha | Number | True |  |  |  |
| Labels |  | Object | True |  |  |  |
|  | Title | TEXT | True |  |  |  |
|  | XLabel | TEXT | True |  |  |  |
|  | YLabel | TEXT | True |  |  |  |
|  | Legend | LEGEND | True |  |  |  |
| BoxStyle |  | Object | True |  |  |  |
|  | Color | String | True |  |  |  |
|  | LineWidth | Number | True |  |  |  |
| Position |  | Object | True |  |  |  |
|  | Position | Array<Number> | True |  |  |  |

#### LINE

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| ColorAndStyle |  | Object | True | 颜色和样式 |  |  |
|  | Color | String | True |  |  |  |
|  | LineStyle |  | True |  |  |  |
|  | LineWidth |  | True |  |  |  |
| Markers |  | Object | True | 标记 |  |  |
|  | Marker | String | True |  |  |  |
|  | MarkerSize | Number | True |  |  |  |
|  | MarkerEdgeColor | String | True |  |  |  |
|  | MarkerFaceColor | String | True |  |  |  |
| Legend |  | Object | True |  |  |  |
|  | DisplayName | String | True |  |  |  |

#### TEXT

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| Text |  | Object | True | 文本 |  |  |
|  | String | String | True |  |  |  |
|  | Color | String | True |  |  |  |
| Font |  | Object | True |  |  |  |
|  | FontName | String | True |  |  |  |
|  | FontAngel | Boolean | True |  |  |  |
|  | FontWeight | Boolean | True |  |  |  |
|  | FontSize | Number | True |  |  |  |
| TextBox |  | Object | True |  |  |  |
|  | Rotation | Number | True |  |  |  |
|  | EdgeColor | String |  |  |  |  |
|  | BackgroundColor | String |  |  |  |  |
|  | LineStyle | String |  |  |  |  |
|  | LineWidth | Number |  |  |  |  |
|  | Margin | Number |  |  |  |  |
| Position |  | Object |  |  |  |  |
|  | Position | Array<Number> |  |  |  |  |
|  | Units | String |  |  |  |  |
|  | HorizontalAlignment | String |  |  |  |  |
|  | VerticalAlignment | String |  |  |  |  |

#### LEGEND

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| PositionAndLayout |  | Object | True | 位置和布局 |  |  |
|  | Location | String | True |  |  |  |
|  | Orientation | String | True |  |  |  |
|  | NumColumns | Number | True |  |  |  |
|  | Position | Array<Number> | True |  |  |  |
|  | Units | String | True |  |  |  |
| Font |  | Object | True |  |  |  |
|  | FontName | String | True |  |  |  |
|  | FontAngel | Boolean | True |  |  |  |
|  | FontWeight | Boolean | True |  |  |  |
|  | FontSize | Number | True |  |  |  |
| ColorAndStyle |  |  | True |  |  |  |
|  | TextColor | String | True |  |  |  |
|  | Color | String | True |  |  |  |
|  | EdgeColor | String | True |  |  |  |
|  | Box | Boolean | True |  |  |  |
| Labels |  |  |  |  |  |  |
|  | Labels | String |  |  |  |  |
|  | TitleString | String |  |  |  |  |
|  | Color | String |  |  |  |  |
|  | FontName | String |  |  |  |  |
|  | FontAngle | Boolean |  |  |  |  |
|  | FontWeight | Boolean |  |  |  |  |
|  | FontSize | Number |  |  |  |  |

### 设置属性接口协议

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 参数 | | 类型 | 是否必需 | 说明 | 示例 | 备注 |
| type |  | String | True | 事件类型 | set\_figure\_attribute | 设置属性 |
| figure\_id |  | Int | False | Figure唯一id | 2254102997960 |  |
| [key]:value |  | Int | False | 属性 | ‘lineStyle’:’--’ |  |
|  | [key] | String | True | 属性名称 | ‘lineStyle’ |  |
|  | value | Any | True | 属性值 | ‘--’ |  |

# 最终交付形态

# 风险点及拟解决方案

## Websocket服务关闭及figure实例的回收

表 5-1端口占用

|  |  |
| --- | --- |
| 描述 | 当断网重连时，websocket链接中断易造成服务端关闭websocket服务并回收figure实例。 |
| 方案 | 设置webscoket心跳和timeout超时时间，规定时间内不关闭服务、不回收figure。重连超过规定时间后将关闭服务并回收figure，并在前端做出提示，引导用户重新进行绘图。 |

## 通信稳定性

表 5-2通信稳定性

|  |  |
| --- | --- |
| 描述 | 由于 WebSocket 连接的建立过程需要双向交互，因此更容易受到网络状况等因素的影响，导致连接中断。 WebSocket 连接则是基于事件驱动的，需要客户端主动向服务器发起连接请求，服务器回应确认连接后，客户端才能与服务器建立连接。 |
| 方案 | 1. 使用连接管理技术：通过使用连接管理技术，例如连接池、心跳包等，可以维护连接的稳定性，防止连接中断。 2. 使用数据打包技术：通过使用数据打包技术，可以将多个数据包打包成一个数据包进行传输，从而提高传输效率，降低网络延迟和抖动。 3. 使用异步通信方式：通过使用异步通信方式，可以将数据传输和用户操作分离，从而减少网络状况对用户体验的影响 4. 使用http通信协议：HTTP 协议是基于请求 - 响应模式的，客户端向服务器发出请求后，服务器立即返回响应，从而使数据传输更加高效。 |

## 网络延迟

表 5-3网络延迟

|  |  |
| --- | --- |
| 描述 | 由于采用服务端渲染模式，目前采用实时渲染的方式不断更新图片，在网络延迟较大时，易造成交互与图片不同步。 |
| 方案 | 参考matlab，只绘制当前可视区域的图形，当交互结束时（例如平移拖动结束、缩放结束）再重新绘制图形。 |

## 同时支持python绘图、julia绘图、python调julia绘图

# 服务端渲染与客户端渲染方案对比

|  |  |  |
| --- | --- | --- |
| 参数 | 服务端渲染（SSR） | 客户端渲染（CSR） |
| **绘图种类完备性** | 由于Matplotlib提供webagg，理论上Matplotlib所有绘图种类都支持 | 需要对Matplotlib和前端绘图库进行桥接转换后才能支持绝大部分绘图 |
| **数据传输** | 通信持久化，通过界面交互不断返回新图片。 | 单次通信，数据初始化时将绘图所需数据一次性传输，之后通信即可关闭 |
| **千万级数据** | 返回数据为图片格式，故不会受数据量影响 | 千万级数据易造成数据传输堵塞或渲染卡顿等问题，需进行特殊处理 |
| **通信模式** | 一对一，即一个绘图窗口对应一个websocket服务 | 一对多，即多个绘图窗口对应一个websocket服务 |
| **网络波动影响** | 由于数据为图片，故网络波动大时仅会造成交互上的卡顿，不会对数据产生影响。 | 若初始化过程中网络波动大，则会造成数据获取失败等问题从而导致绘图失败。若初始化结束后网络波动大，由于是客户端渲染绘图，网络波动对其毫无影响。 |
| **扩展性** | 不需要针对绘图类型进行额外开发，可针对交互进行扩展 | 需要对每个绘图种类进行桥接转换开发，可针对交互进行扩展 |
| **开发难易度** | 中：主要集中在数据通信和页面交互 | 高：不仅需要开发数据通信和页面交互，还需耗费大量时间处理桥接转换 |

表 5-1