绘

图

工

具

# 绘图（ZBase）

绘图的基础类，职责是只负责画图与绘图相关数据处理，不会处理图元之间的关系。

**MYREGISTER**(类名)注册反射(必须注册)。

通过***setAttrs***函数可以设置所有属性,子类通过重载***attrUpdate***函数可以获取自己类独有的属性。

通过重载***paintOver***函数绘制图形。

支持异步和同步处理数据、可以限定边界大小(但是必须设定SceneSize和Flags必须有|QGraphicsItem::ItemSendsScenePositionChanges)

## 虚函数

| 序号 | 操作标识 | 功能描述 |
| --- | --- | --- |
|  | getClassName | 得到类名，此函数必须重载，是返回类名即可 |
|  | paintOver | 重画函数，在此函数中绘制图元 |
|  | attrUpdate | 更新属性,设置属性的时候会调用的函数，一般是数据的来源 |
|  | getAttrUpdate | 得到属性的时候要调用的函数,用来更新属性,调用**getAttrs**函数的时候会调用 |
|  | Asynchronous | 子类实现，根据***upDateData***函数的第二次参数，选择是否异步处理函数  此函数会自动调用，在子类中实现自己的异步（或者同步）操作，  会在处理数据后，会刷新图元（一般用来实现自己类数据设定、数据处理），一般和工具类交互使用,通过***getDateData***函数得到处理后的数据 |
|  | isContainPos | 是否包含点击的点,如果没有包含就不接受鼠标事件(会在调用items()后使用) |
|  | setEdit | 设置编辑的状态 |
|  | selectdChanged | 选择改变 |
|  | asynFinish | 同步或者异步数据处理完成 |
|  | AddScene | 但添加到场景的时候会触发 |
|  | getThis | 得到this指针,在多重继承的时候要重载 |
|  | getKeyType | 得到属性key值的类型 |

## 成员函数

| 序号 | 操作标识 | 功能描述 |
| --- | --- | --- |
|  | setAttrs | 设置所有属性，会触发***attrUpdate***函数 |
|  | setAttr | 设置key对应的属性,通过第三个决定是否触发***attrUpdate***函数 |
|  | getAttrs | 得到属性,会触发***getAttrUpdate***函数 |
|  | getDateData | 得到更新后的数据 |
|  | setBounding | 设置图元的范围,会触发更新图元 |
|  | setSelected | 设置是否选中,但是flgs()值中要设置是否能够选中即(QGraphicsItem::ItemIsSelectable) |
|  | setId | 设置标识 |
|  | getId | 得到标识 |
|  | sendAttrChanged | 发送属性改变信号 |
|  | setItemType | 设置图元的类型 |
|  | getItemType | 得到图元的类型 |
|  | isAttrValueValid | 判断一个属性值是否有效 |

# 工具（CZBaseTool）

处理图元之间的关系、图元和视图的事件、图元的变化（例如大小、选中）。

**MYREGISTER**(类名)注册反射(必须注册)。

通过设置活跃的状态要决定是否要传递事件

## 虚函数

事件的虚函数见CZBaseTool.h，item事件函数是以item开头，view事件是以view开头，

如果重载的函数返回true,则事件不向下传递，事件传递的顺序和创建tool的先后相关。

| 序号 | 操作标识 | 功能描述 |
| --- | --- | --- |
|  | getClassName | 得到类名，此函数必须重载，是返回类名即可 |
|  | upDataTool | 更新工具，从此函数中得到要改变的操作的图元,***initTool***函数会调用,在管理类也会调用 |
|  | initTool | 初始化工具,默认调用upDataTool(CZBase \*item);会把所有的item传递 |
|  | addItem | 向工具中添加一个图元，在加入操作容器后触发此函数 |
|  | removeItem | 把一个图元从工具中移除 |
|  | dealData | 处理数据 |
|  | setEditItem | 设置编辑的item |

## 成员函数

| 序号 | 操作标识 | 功能描述 |
| --- | --- | --- |
|  | setOperationItems | 设置被操作的item的集合,原来的会被清除 |
|  | addOperationItem | 添加要操作的item，会触发addItem |
|  | removeOperationItem | 移除要操作的item,会触发removeItem |
|  | getItemById | 得到操作item(只属于该工具)通过id |
|  | getOperationItems | 得到使用操作的图元 |
|  | setDealData | 设置要处理的数据 |
|  | setToolActiva | 设置是否激活 |
|  | getToolActiva | 得到是否激活 |
|  | isBelong | 判断是否属于该工具 |

# 总管理类（CManagement）

此类负责和外界交互。

文件操作说明:可以通过配置文件加载所需要的工具，通过配置文件配置图元对应的工具，通过配置文件创建图元。也可以通过函数动态创建。可以把工具配置，图元对应工具、图元属性保存到配置文件中。

图元操作说明:创建、获取、删除、编辑图元。得到、修改图元的属性

工具说明:可以创建一个工具，配置一个类型对应的工具，传递数据并得到处理后的结果，控制工具的是否活跃。

视图说明:可以设置和获取视图属性

## 成员函数

| 序号 | 操作标识 | 功能描述 |
| --- | --- | --- |
| 文件操作 | | |
|  | loadTool | 加载工具 |
|  | saveTool | 保存工具的配置文件 |
|  | configItemTool | 配置item的工具,此函数要位于loadTool(QString fileName);之后 |
|  | saveConfigItemTool | 保存配置item的工具 |
|  | loadItemFileName | 加载图元配置文件  调用工具管理:***initAllToo***l函数  调用工具:***initTool***函数 |
|  | saveItemFile; | 保存item的属性到配置文件 |
| 图元 | | |
|  | createItem | 创建一个绘图的item(如果有父类，父类必须先创建),  调用图元管理：createItem函数  调用图元：***setAttrs***、***AddScene***函数  调用工具：***addOperationItem***函数  发出sig\_CreateFinish信号 |
|  | createItemByType | 通过类型创建一个item |
|  | deleteItem | 删除item，会设置编辑状态为不编辑，  调用工具：***removeItem***函数 |
|  | clearAllItem | 清除所有的图元  调用工具：***removeItem***函数 |
|  | setCurrentId | 设置当前图元 |
|  | getCurrentId | 得到当前item的id |
|  | getSelects | 得到选中的Id |
|  | setEdit | 设置图元的编辑状态，会调用对应图元***setEdit***函数，工具***setEditItem***函数 |
|  | getEdit | 是否编辑 |
|  | getKeyType | 得到一个键值的类型,调用图元***getKeyType***函数 |
| 图元属性 | | |
|  |  |  |

# 管理绘图（CManagementItem）

此类管理通过改类创建的所有图元，能够创建图元、删除图元、选中图元、编辑图元、获取图元、得到当前图元、设置和获取视图属性。

## 成员函数

| 序号 | 操作标识 | 功能描述 |
| --- | --- | --- |
|  | createItem | 创建一个绘图的图元 |