整体框架

画图中心面板，添加几何图形

Cell4

Cell3

Cell2

Cell1

Cell1-child2

Cell1-child1

Cell2-child2

Cell2-child1

Cell3-child2

Cell3-child1

Cell4-child2

Cell4-child1

任务：每点击单个构元

Cell1，或者其孩子

Cell1-child1，就需要在

中心worktabs中显示

主要讨论worktabs和celltabs的多窗口显示

Treewidget的结构：

它根据view里面gds的信息获取构元名字的信息，并将构元名字信息以树状结构将显示到treewidget控件里面，也就是treewidget根据view里面的信息来显示，view1对应一个treewidget1，view2对应treewidget2.....treewidget里面的树状层次结构反映的是构元的引用与被引用关系，现在只显示了两层，有可能有3层，4层，....等

根据主程序传来的filename路径名，获取gds文件数据，并根据读

取的数据在view上面画gds整体图形，这整个功能在view的loadfile函数实现。

View类的功能

可以在view上面添加线，矩形等几何图

将读取的gds文件中的构元信息提取出来保存在链表cell中，并传递给主程序中的cellDockView()函数，并在该函数中新建一个treewidget来显示cell信息，将treewidget加入celltabs进行管理

打开gds文件的过程：点击open动作

上面的思想解决了：可以打开多个gds文档，并分别在多个view中显示整体图，在多个treewidget中显示构元信息，并且treewidget信息来自view。

问题：1.view是由worktabs管理，treewidget是由celltabs管理，worktabs和celltabs相互独立，如何让它们同步显示一个文件，例如，当worktabs中view3删除了，怎样使对应的celltabs中的treewidget3也跟着删除，

2，如果点击treewidgetN中某个构元cell1，怎样将单个构元cell1在worktabs中显示出来。我的思想是，要显示单个构元，必须新建一个viewcell1(cell1是构元名字)，在viewcell1中画出该构元，并将viewcell1添加到worktabs，设置worktabs的当前显示为viewcell1（难点：treewidgetN中的cell1只是一个名字，并不包含怎么画，若要在新建的viewcell1中画出，必须要找到treewidgetN对应的viewN中gdsN的信息，怎样将viewN中gdsN的信息传递给viewcell1并显示，每个gdsN信息都保存在每个viewN中）

celltabs

worktabs

新建的treewidget1层次结构，是由view1里面的构元cell信息确定，这样就实现了每打开一个gds2，就新建一个view2来画gds2，同时返回gds2中cell2信息，新建一个treewidget2来显示cell2信息

根据view1里面的构元cell信息，创建treewidget1层次结构

新建的treewidget1

执行loadfile（filename）函数画出整体图形,同时返回构元cell的信息

在主程序中新建一个view1,执行

同时创建一个

Treewidget1

新建的view1

执行fileopen函

数获取路径名filename

主程序

我觉得还是上次我说的那个思路。

我描述一下怎么实现

1. 在view调用cellDockView()函数生成treewidget时把view本身的引用传递给新生成treewidget，这样treewidget就能访问关联的view里面的cell数据。
2. Treewidget类里面你需要定义一个通知消息接口，假设为ActionHandle()，这样view里面新增/删除/修改cell等数据时。调用Treewidget:: ActionHandle()要求他更新cell列表。该通知消息也可以告知要Treewidget要求删除自身。不过我觉得删除应该是直接要求cellTabs删除指定的Treewidget(生成的时候你应该知道要删除的Treewidget相关数据，如在cellTabs中的索引，以及它的引用)
3. Treewidget接收到要求更新的消息后，通过保存的view引用读取数据并更新。
4. 因为Treewidget有对应的view的引用。如果选取某个cell，你应该能从cell链表中找到相关的所有数据。新建cellview时，就能把这些数据传递给它，新建的cellview就能画出相应的数据。

不过我觉得这种构架不是最好的，我觉得如果把文档数据与view分离出来会更好，这样灵活性就很大，根据你的说明是一个Treewidget对应多个view的构架。下面是我的思路。

1. 文档数据的读取保存通过CGdsDocument类来读取和保存。
2. 主程序或者主框架调用CGdsDocument::LoadFile()读取GDS文件，提取必要的信息。构建cell链表。
3. 主程序创建gdsview1和Treewidget1，同时CGdsDocument的引用传递给他们。然后把gdsview1添加到worktabs。gdsview1根据document里面的数据画出图形，Treewidget1则构建cell数据，cell数据都跟document里面的链表数据一一对应。
4. 主程序把生成gdsview1和Treewidget1的引用也保存在Document中。
5. 当gdsview1中修改了数据，数据保存到Document后，然后要求主程序刷新相应Document中的Treewidget1数据，(当然只刷新必要的数据，不需要全部重新装载，否则效率低下)。当然如果有跟该文档相关的其他view，则也需要刷新，具体情况见后面的同步全部数据显示描述。
6. 如果关闭gdsview1时，可以要求主程序删除相应Document中Treewidget1，不过如果一个Document有多个view(包含多个cell-view和一个gdsview)，则只有最后的view关闭后才能删除
7. Treewidget1点击某个cell或者cell-child要求显示时，你知道该cell的索引信息。然后要求主程序生成一个新cell-view，传递cell索引信息和Document引用信息。生成后把cell-view添加到worktabs中。也把新生成的cell-view引用添加到document中。
8. cell-view根据document和cell索引信息从document里面读取相应的信息并画出来。
9. 当cell-view中进行图形绘制等操作时，把数据保存到document中，然后要求主程序同步所有数据。主程序根据document保存的所有view以及Treewidget，分别调用他们的刷新函数来同步数据。
10. 我个人认为没有必要采用不同的cell-view，cell-view就采用gdsview，只是把gdsview放大到让操作的cell显示到整个客户区，给人的感觉几乎就是显示相应的cell。