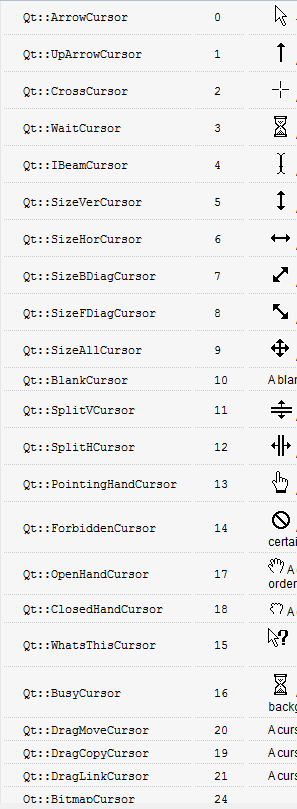
<http://blog.csdn.net/afterward___/article/details/46408355>

QT提供了十分便捷的设置鼠标形状的方法，在QT界面的根类QWidget中有QCursor cursor(); void setCursor(QCursor &); void unsetCursor()三个方法，因而对于一切直接或间接继承自QWidget的控件都可以设置鼠标进入控件范围的鼠标形状。

其次，就QCursor类而言，QT提供了两种类型的鼠标形状：其一是内置鼠标形状（如下）【例如setCursor(Qt::ArrowCursor)设置鼠标为箭头形状】



其二是图片自定义鼠标形状

http://img.blog.csdn.net/20150608095152444?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkbi5uZXQvYWZ0ZXJ3YXJkX19f/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/Center

相应的可以设置图片上热点的位置（即具体产生点击效果的图片位置，当hotX和hotY均为-1时，默认是图片的正中心为热点）。【例如setCursor(QPixmap("\*.png"))】

**分享一下本人遇到的问题：在做一个画图工具（如下图）的时候，希望点击工具栏上不同的按钮后，进入下面的画布（QGraphicsView）就将鼠标切换成相应的形状；当鼠标悬浮在画布上的图元（QGraphicsItem）上时相应的也形状，如悬浮在直线上则显示http://img.blog.csdn.net/20150608100150601?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkbi5uZXQvYWZ0ZXJ3YXJkX19f/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/Center表示可以左右拖动拉伸缩短等。**



我的做法是继承了QGraphicsItem来绘制相应的图形，同时重写了QGraphicsItem的hoverEnterEvent函数，在函数中setCursor为目标形状；同时继承QGraphicsView类，当点击了工具栏上相应按钮后就在此类中直接setCursor为目标形状。奇怪的是，**在继承QGraphicsView类中切换鼠标形状总失败，尽管setCursor();语句也执行了**。

解决方法就是**在QGraphicsView子类中使用viewport()->setCursor()而不是直接的setCursor()，这样才能真正改变视觉上的鼠标形状。viewport()函数定义在QAbstractScrollArea类中，QGraphicsView继承自QAbstractScrollArea类，对于更新更新视图内容应该用viewport()->update()，而不是直接用update()。**