

# 绘图设备

---

绘图设备是指继承**QPaintDevice**的子类，你可以使用**QPainter**直接在其上面绘制图形，Qt一共提供了四个这样继承**QPaintDevice**的绘图设备类，分别是**QPixmap**、**QBitmap**、**QImage**和 **QPicture**。

常用**QPixmap**屏幕上显示图片，**QImage**传输数据

- **QPixmap**专门为图像在屏幕上的显示做了优化
- **QBitmap**是**QPixmap**的一个子类
- **QImage**专门为图像的像素级访问做了优化
- **QPicture**则可以记录和重现**QPainter**的各条命令

## QPixmap

---

- **QPixmap**可以接受一个字符串作为一个文件的路径来显示这个文件
- 使用**QPainter**的**drawPixmap()**函数可以把这个文件绘制到一个**QLabel**、**QPushButton**或者其他设备上。
- **QPixmap**是针对屏幕进行特殊优化的，因此，它与实际的底层显示设备息息相关，是操作系统提供的原生的绘图引擎。所以，在不同的操作系统平台下，**QPixmap**的显示可能会有所差别。
- **QPixmap**主要是用于绘图，针对屏幕显示而最佳化设计，**QImage**主要是为图像I/O、图片访问和像素修改而设计的

## QBitmap

---

- **QBitmap**继承自**QPixmap**，主要用于显示单色位图。是**QPixmap**子类，因此具有其所有特性。**QBitmap**的色深始终为**1**。
- 色深这个概念来自计算机图形学，是指用于表现颜色的二进制的位数。我们知道，计算机里面的数据都是使用二进制表示的。为了表示一种颜色，我们也会使用二进制。**1**个位只有两种状态：**0**和**1**，因此它所表示的颜色就有两种，黑和白。所以说，**QBitmap**实际上是只有黑白两色的图像数据。
- 由于**QBitmap**色深小，因此只占用很少的存储空间，所以适合做光标文件和笔刷。

# QImage

---

- **QPixmap**使用底层平台的绘制系统进行绘制，无法提供像素级别的操作，而**QImage**则是使用独立于硬件的绘制系统，实际上是自己绘制自己，因此提供了像素级别的操作，并且能够在不同系统之上提供一个一致的显示形式。
- 声明了一个**QImage**对象，大小是**3 x 3**，颜色模式是**RGB32**，即使用**32**位数值表示一个颜色的**RGB**值，也就是说每种颜色使用**8**位。然后我们对每个像素进行颜色赋值，从而构成了这个图像
- **QImage**可通过**setPixel()**和**pixel()**等方法直接存取指定的像素。

```
1 QImage image(3,3,QImage::Format_RGB32);  
2 value=qRgb(186,147,39); 0xffbb8372 //32位表示一个颜色  
3 image.setPixel(1,1,value);
```

- 可以把**QImage**想象成一个**RGB**颜色的二维数组，记录了每一像素的颜色。
- 当图片较大时，我们可以先通过**QImage**将图片加载进来，然后把图片缩放成需要的尺寸，最后转换成**QPixmap** 进行显示。