<http://blog.csdn.net/qq_16952303/article/details/51585577>

看了下Qt的帮助文档，发现connect函数最后还有一个缺省参数.

connect函数原型是这样的：

QMetaObject::Connection QObject::connect(const QObject \* sender, const char \* signal, const QObject \* receiver, const char \* method, Qt::ConnectionType type = Qt::AutoConnection)

最后的ConnectionType是一个缺省的参数，默认为自动连接方式，我们来看一下这个参数有哪些值

Qt supports these signal-slot connection types:  
  
Auto Connection (default) If the signal is emitted in the thread which the receiving object has affinity then the behavior is the same as the Direct Connection. Otherwise, the behavior is the same as the Queued Connection."

Direct Connection The slot is invoked immediately, when the signal is emitted. The slot is executed in the emitter's thread, which is not necessarily the receiver's thread.

Queued Connection The slot is invoked when control returns to the event loop of the receiver's thread. The slot is executed in the receiver's thread.

Blocking Queued Connection The slot is invoked as for the Queued Connection, except the current thread blocks until the slot returns.  
Note: Using this type to connect objects in the same thread will cause deadlock.  
  
  
Unique Connection The behavior is the same as the Auto Connection, but the connection is made only if it does not duplicate an existing connection. i.e., if the same signal is already connected to the same slot for the same pair of objects, then the connection is not made and connect() returns false

可以看出一共是5个值:自动、直接、队列、阻塞队列、唯一。

我们先看下 直接和队列。

直接连接的大概意思是：信号一旦发射，槽立即执行，并且槽是在信号发射的线程中执行的。

队列连接的大概意思是：信号发射后当事件循环返回到接收线程时槽函数就执行了，也就是说这种连接方式不是立即触发槽函数的，而是要排队等的，并且是在槽函数的线程中执行。

再来看看 自动和阻塞队列方式

自动连接的大概意思是：信号发射对象如果和槽的执行对象在同一个线程，那么将是直连方式，否则就是队列方式。

阻塞队列方式：在槽函数返回之前槽函数所在的线程都是阻塞的。

唯一方式：和直连相同，但是只能一对一连接。

Qt帮助文档中提醒到:Be aware that using direct connections when the sender and receiver live in different threads is unsafe if an event loop is running in the receiver's thread, for the same reason that calling any function on an object living in another thread is unsafe.

即跨线程调QObject对象是不安全的。

Call qRegisterMetaType() to register the data type before you establish the connection。

另外信号槽的参数必须是注册的MetaType，所以当你使用自定义的类型或者 没有注册的类型，都要调用qRegisterMetaType()进行注册,因为Qt需要保存你的参数。

版权声明：本文为博主原创文章，未经博主允许不得转载。