使用多线程功能实现定时器

1、UITaskDispatcher：绑定到应用主线程的专有任务分发器， 由Ability执行getUITaskDispatcher()创建并返回得到。 由该分发器分发的所有的任务都是在主线程上按顺序执行，它在应用程序结束时被销毁。

TaskDispatcher uiTaskDispatcher = getUITaskDispatcher();

2、delayDispatch

异步延迟派发任务：异步执行，函数返回值为Revocable，内部会在延时指定时间后将任务派发到相应队列中。延时时间参数仅代表在这段时间以后任务分发器会将任务加入到队列中，任务的实际执行时间可能晚于这个时间。具体比这个数值晚多久，取决于队列及内部线程池的繁忙情况。

3、Revocable

取消任务：Revocable是取消一个异步任务的接口。异步任务包括通过 asyncDispatch、delayDispatch、asyncGroupDispatch 派发的任务。如果任务已经在执行中或执行完成，则会返回取消失败。

调用revocable的revoke()方法取消一个异步任务

Boolean revoked = revocable.revoke();

4、ToastDialog

ToastDialog是在窗口上方弹出的对话框，是通知操作的简单反馈。ToastDialog会在一段时间后消失，在此期间，用户还可以操作当前窗口的其他组件。

创建和使用：

new ToastDialog(getContext())

.setText("This is a ToastDialog")

.setAlignment(LayoutAlignment.CENTER)

.show();

