

Android的联通性---Wi-Fi Direct (一)

2012-11-22 20:52

5803人阅读

评论(0)

收藏

举报

分类: 学习笔记 (426)

本文译自: <http://developer.Android.com/guide/topics/connectivity/wifi2p.html>

Wi-Fi Direct允许Android 4.0 (API Level 14) 以后的设备, 使用相应的硬件通过Wi-Fi直接的彼此相连, 而不需要中间访问点。当每个设备支持Wi-Fi Direct时, 使用这些API就能够发现并连接另一个对等设备, 而且通信距离要远远超过蓝牙连接。这对于要在用户间共享应用程序的数据是有用的, 如多人游戏或图片共享应用程序等。

Wi-Fi Direct由以下主要部分组成:

1. 在WifiP2pManager类中定义了用于彼此对等发现、请求、连接的方法;
2. 用于获取WifiP2pManager方法调用结果 (成功或失败) 通知的监听器。当调用WifiP2pManager方法时, 每个方法都能够接收一个作为参数传入的特殊监听器;
3. 用于通知你Wi-Fi Direct框架所检测到的特定事件的Intent对象, 如删除链接或发现新设备。

这三个主要的API组件经常要一起使用。例如, 给discoverPeers()方法提供一个WifiP2pManager.ActionListener监听器, 以便能够用ActionListener.onSuccess()和ActionListener.onFailure()方法来获取通知。如果discoverPeers()方法发现对等设备列表发生改变, 还会发出一个WIFI_P2P_PEERS_CHANGED_ACTION类型的Intent广播。

API概要

WifiP2pManager提供了一些用于跟设备的Wi-Fi硬件相互作用的方法, 这些方法完成设备的彼此发现和连接。以下是操作是有效的:

表1.Wi-Fi Direct方法

方法	介绍
<code>initialize()</code>	把应用程序注册到Wi-Fi框架中, 它必须在调用其他Wi-Fi Direct方法之前调用。
<code>connect()</code>	用指定的配置来启动设备间的对等连接。
<code>cancelConnect()</code>	取消任何进行中的对等设备间连接请求。
<code>requestConnectInfo()</code>	请求设备的连接信息。
<code>createGroup()</code>	用当前设备作为组管理员来创建一个对等组。

- removeGroup() 删除当期对等设备组。
- requestGroupInfo() 请求对等组信息。
- discoverPeers() 启动对等点的发现。
- requestPeers() 请求当前发现的对等点的列表。

WifiP2pManager类的方法会让你传入一个监听器，以便Wi-Fi Direct框架能够通知你的Activity该调用的状态。下表介绍了WifiP2pManager类的方法调用时可用的监听器接口：

表2.Wi-Fi Direct监听器

监听器接口	关联的操作
WifiP2pManager.ActionListener	connect(),cancelConnect(),createGroup(),removeGroup(), and discoverPeers()
WifiP2pManager.ChannelListener	initialize()
WifiP2pManager.ConnectionInfoListener	requestConnectInfo()
WifiP2pManager.GroupInfoListener	requestGroupInfo()
WifiP2pManager.PeerListListener	requestPeers()

Wi-Fi Direct API还定义了一些在某些Wi-Fi Direct事件发生时，用广播的形式发出的Intent对象，如在发现对等设备或设备的Wi-Fi状态发生变化时。你能够在你的应用程序中注册接收这些Intent对象，并通过[创建广播接收器](#)来处理这些Intent对象：

表3.Wi-Fi Direct Intent

Intent	介绍
WIFI_P2P_CONNECTION_CHANGED_ACTION	在设备的Wi-Fi连接状态变化时，发出这个广播。
WIFI_P2P_PEERS_CHANGED_ACTION	在调用discoverPeers()方法时，发出这个广播，如果你要在应用程序中处理这个Intent，通常是希望调用requestPeers()方法来获取对等设备的更新列表。
WIFI_P2P_STATE_CHANGED_ACTION	当启用或禁用设备上的Wi-Fi Direct时，发出这个广播。
WIFI_P2P_THIS_DEVICE_CHANGED_ACTION	当设备的细节（如设备的名称）发生变化时，发出这个广播。

WIFI_P2P_THIS_DEVICE_CHANGED_ACTION 接收的细节（如接收的名称）及更改化时，及山达！/ 猫。
ICE_CHANGED_ACTION
ON

给Wi-Fi Direct的Intent创建广播接收器

广播接收器允许你接收由Android系统发出的Intent广播，以便你的应用程序能够响应你感兴趣的事件。以下是创建处理Wi-Fi Direct的Intent接收器的基本步骤：

1. 创建一个继承BroadcastReceiver类的类。这个类的构造器，要有WifiP2pManager、WifiP2pManager.Channel和Activity类型的参数。这样就允许广播接收器把更新发送给Activity以及要访问的Wi-Fi硬件和通信通道。
2. 在广播接收器的onReceive()方法中检查你感兴趣的Intent对象，根据接收到的Intent来执行必要的操作。例如，如果广播接收器接收到一个WIFI_P2P_PEERS_CHANGED_ACTION类型的Intent，就可以调用requestPeers()方法来获取当前发现的对等设备的列表。

下列代码显示了如何创建一个典型的广播接收器。该广播接收器需要一个WifiP2pManager对象和一个Activity对象作为参数，并且使用这两个类在广播接收器接收到Intent时，执行合适的需要的操作：

```
/**  
  
 * A BroadcastReceiver that notifies of important Wi-Fi p2p events.  
  
 */  
  
public class WifiDirectBroadcastReceiver extends BroadcastReceiver {  
  
    private WifiP2pManager manager;  
  
    private Channel channel;  
  
    private MyWifiActivity activity;  
  
    public WifiDirectBroadcastReceiver(WifiP2pManager manager, Channel channel,  
        MyWifiActivity activity) {
```

```

super();

this.manager = manager;

this.channel = channel;

this.activity = activity;
}

@Override

public void onReceive(Context context, Intent intent) {

    String action = intent.getAction();

    if (WifiP2pManager.WIFI_P2P_STATE_CHANGED_ACTION.equals(action)) {

        // Check to see if Wi-Fi is enabled and notify appropriate activity

    } else if (WifiP2pManager.WIFI_P2P_PEERS_CHANGED_ACTION.equals(action)) {

        // Call WifiP2pManager.requestPeers() to get a list of current peers

    } else if (WifiP2pManager.WIFI_P2P_CONNECTION_CHANGED_ACTION.equals(action))
    {

        // Respond to new connection or disconnections

    } else if (WifiP2pManager.WIFI_P2P_THIS_DEVICE_CHANGED_ACTION.equals(action)) {

        // Respond to this device's wifi state changing

    }

}

}

```