

# Comunicación Mediante Wi-Fi Direct

Ayala Carlos, Cadena Paul, Chicaiza Christian, Miranda Alexis Grupo 1

February 16, 2017

## Abstract

Mediante el uso de Wi-Fi Direct es posible establecer comunicación entre varios dispositivos siempre y cuando estos posean esta tecnología, en la actualidad varios dispositivos cuentan con la tecnología de Wi-Fi Direct como lo son: smartphones, tablets, Smart tvs, impresoras, entre otros. El uso de Wi-Fi Direct tiene grandes ventajas como: el que no se necesita de un Access point al cual los dispositivos deban conectarse y la velocidad de transferencia es más rápida que bluetooth. La aplicación creada llamada “Chat Fis” fue desarrollada en Android Studio el cual es el entorno de desarrollo integrado oficial para la plataforma Android en el cual mediante programación orientada a objetos usando clases en java y XML concluyendo así en una aplicación la cual funciona de dos formas : la primera como chat one-to-one entre varios dispositivos con un máximo de 20 y la segunda como intermediario entre la comunicación de dos dispositivos.

## 1 Introducción

En la actualidad el uso de dispositivos móviles, especialmente smartphones y tablets abarcan un gran sector. Debido a esto cada vez surgen más tecnologías y más aplicaciones para estos dispositivos. Los usuarios de estos dispositivos cada vez tienen la necesidad de estar conectados con otros usuarios u otros dispositivos, para lo cual han surgido nuevas tecnologías como el Wi-Fi Direct el cual nos permite conectar dispositivos cercanos sin la necesidad de un punto de acceso a internet. Estas conexiones se llevan a cabo mediante la creación de redes de tipo P2P o Ad-Hoc las cuales nos permiten el intercambio de información entre los dispositivos que la conforman. Actualmente, la gran mayoría de smartphones Android incorporan la

tecnología de Wi-Fi Direct el cual se incluyó por primera vez en Android 4.0 Ice Cream Sandwich. Lógicamente aquellos teléfonos con versión inferior de Android no soportan Wi-Fi Direct. Esta tecnología permite la conexión de dispositivos entre si mediante la creación de una red P2P, con la única condición de tener habilitada la red Wi-Fi. Debido al masivo uso del S.O.

Android la implementación de una aplicación la cual utilice esta tecnología resulta ser muy útil.

Y gracias al wifi direct se ha podido implementar una aplicación de chat simultaneo entre celulares llamadi Wi-Fi Direct Multichat que no necesariamente deben estar conectados al internet si no simplemente por una conexión de wifi-direct entre ellos.

Wi-Fi Chat-Fis Messenger es una aplicación demo Android's que intenta superar algunas limitaciones de Wi-Fi Direct. Por el momento, las aplicaciones convencionales no permite que un dispositivo participe simultáneamente en dos grupos de Wi-Fi Direct.

Esta aplicación trata de superar esta limitación, de hecho, un usuario puede comunicarse con un gran número de personas cercanas sin conexión a Internet, utilizando las comunicaciones punto a punto. El objetivo principal es la gestión de múltiples chats, cola de mensajes enviados cuando la conexión no está disponible y enviarlos al mismo tiempo tan pronto como sea posible.

## 2 Teoría

Wi-Fi Direct es una nueva tecnología definida por Wi-Fi Alliance en la que los dispositivos capaces se pueden conectar directamente entre sí de forma rápida, segura y conveniente para realizar tareas como la impresión, sincronización y uso compartido de datos. En este artículo ofrecemos una visión completa de las funcionalidades definidas en la Especificación Directa Wi-Fi junto con detalles del protocolo subyacente. Este documento ofrece una visión general de la especificación Wi-Fi Direct, centrándose en sus nuevas funcionalidades e ilustra tres procedimientos representativos de formación de grupos junto con los mecanismos de ahorro de energía involucrados. Más

de una década después de su diseño inicial, el estándar IEEE 802.11, se ha convertido en una de las formas más comunes de acceder a Internet. Wi-Fi tiene su presencia en muchos tipos de dispositivos como teléfonos inteligentes, TV, impresoras, automóviles, asistencia sanitaria, etc. Durante mucho tiempo el wi-fi se limitó al modelo básico de puntos de acceso que crean redes inalámbricas y dispositivos de estación que se conectan a redes inalámbricas. Wi-Fi Direct permite a los dispositivos comunicarse directamente entre sí utilizando métodos similares al Wi-Fi tradicional, excepto que no requieren el uso de un punto de acceso central. En su lugar, los

dispositivos utilizan un "punto de acceso de software" (Soft AP). La conectividad directa de dispositivo a dispositivo ya era posible en el estándar original IEEE 802.11 mediante el modo de operación ad-hoc. Sin embargo,

esto nunca fue capaz de marcar su presencia en el mercado debido a varios inconvenientes o limitaciones en los requisitos, p. La falta de ahorro de energía eficiente apoyo o extendido QoS capacidades. El último avance relacionado con el dispositivo Wi-Fi al espacio de comunicaciones del dispositivo es 802.11z, también conocido como TDLS (Tunneled Direct Link Setup), que permite la comunicación directa entre dispositivo y dispositivo, pero requiere que las estaciones estén asociadas con el mismo AP.

La tecnología Wi-Fi Direct, tal como se describe en "Especificación técnica Wi-Fi Peer-to-Peer (P2P)" adopta un enfoque diferente para mejorar la conectividad entre dispositivos. En lugar de aprovechar el modo de operación ad-hoc, Wi-Fi Direct se basa en el modo de infraestructura IEEE 802.11 con éxito y permite que los dispositivos negocien quién asumirá las funciones de tipo AP. Por lo tanto, permite que los dispositivos Wi-Fi heredados se conecten a la red Wi-Fi Direct que no hubiera sido posible de otra manera.

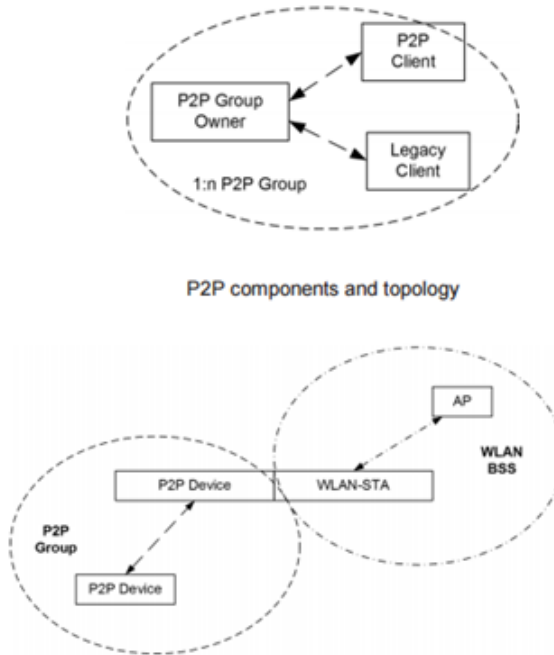
#### Arquitectura

Los dispositivos Wi-Fi Direct, formalmente conocidos como dispositivos P2P, se comunican estableciendo Grupos P2P, que son funcionalmente equivalentes a las redes de infraestructura Wi-Fi tradicionales. El dispositivo que implementa AP como funcionalidad en el Grupo P2P se conoce como el propietario del grupo P2P (P2P GO), y los dispositivos que actúan como clientes se conocen como clientes P2P.

La naturaleza lógica de los roles P2P admite diferentes implementaciones arquitectónicas; Uno de esto se ilustra en la Fig. 1 representa un escenario con dos grupos P2P. El primer escenario es un teléfono móvil que comparte su conexión 3G con dos portátiles; En este primer escenario, los tres dispositivos forman un grupo, el teléfono está actuando como P2P GO mientras que los dos portátiles se comportan como clientes P2P. Con el fin de extender la red, una de las computadoras portátiles establece un segundo grupo P2P con una impresora; Para este segundo grupo, el portátil actúa como P2P GO. Con el fin de actuar tanto como cliente P2P y como P2P GO la computadora portátil suele alternar entre las dos funciones de tiempo compartido de la interfaz Wi-Fi. Al igual que un AP tradicional, un P2P GO

se anuncia a través de balizas que contienen elementos de información P2P adicionales. P2P IE está incluido en todos los marcos de administración. Los dispositivos heredados ignoran estos elementos de información y marcos de acción. La Especificación Directa Wi-Fi requiere que el dispositivo P2P que se convierte en el propietario del grupo también debe proporcionar la aplicación del servidor DHCP en su sistema [3] para proporcionar a los clientes P2P direcciones IP. Además, sólo se permite al P2P GO conectar

los dispositivos de su Grupo P2P a una red externa. Por último, Wi-Fi Direct no permite transferir el papel de P2P GO dentro de un Grupo P2P. De esta manera, si el P2P GO abandona el Grupo P2P, entonces el grupo es derribado, y tiene que ser restablecido usando algunos de los procedimientos especificados.



**Fig.1 Wi-Fi Direct supported topologies**

### 3 Implementación

#### 3.1 Diseño

Es importante tener en cuenta que Wi-Fi Direct ha sido diseñado para soportar la comunicación D2D dentro de un grupo, sin embargo su protocolo no impide la comunicación entre diferentes grupos. De hecho, un dispositivo puede actuar como un puente entre dos grupos, o entre el grupo y otras redes.



Communication between two Wi-Fi Direct groups.

Fig. 2:

Así pues se tiene que la aplicación funcionara de dos modos uno en el que un dispositivo puede conectarse con varios dispositivos

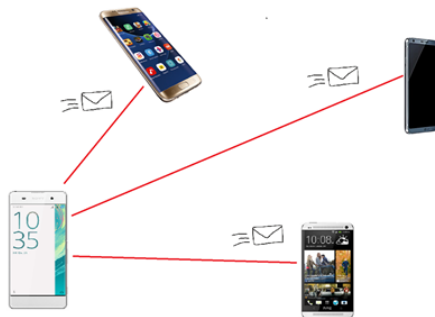


Fig. 3: Dispositivo conectado con varios dispositivos

Y otro en la que usando tres dispositivos podremos enviar un mensaje desde un dispositivo A hacia un dispositivo C pasando por un dispositivo B



Fig. 4 Un dispositivo conectado con segundo y este con un tercer dispositivo

### 3.2 Desarrollo

Permisos que permiten acceder y modificar el estado de las conexiones WI-FI y el estado de la red.

```
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

Fig. 5 Permisos android

Antes de poder realizar las funcionalidades que ofrece la API Wifi.p2p es necesario obtener una instancia de la clase WifiP2pManager, que es la clase que contiene los métodos para descubrir pares y conectarse con los dispositivos entre otros, y a continuación crear un objeto BroadcastReceiver.

```
WifiP2pManager mManager =(WifiP2pManager)getSystemService(Context.WIFI_P2P_SERVICE);
Channel mChannel = mManager.initialize(this, getMainLooper(), null
BroadcastReceiver mReceiver = new WifiDirectBroadcastReceiver(mManager, mChannel, this);
```

Fig. 6 Instanciacion clase Wifip2p

MainActivity.java

```
Log.d(TAG, "handleMessage, p2pDevice created with: " + p2pDevice.deviceName + ", "
      + p2pDevice.deviceAddress + ", " + device.getDestinationIpAddress());
manageAddressMessageReception(device);
String TablaRouting[][] = new String[20][2];
for(int i=0;i<20;i++){
    TablaRouting[i][0]=p2pDevice.deviceName;
    TablaRouting[i][1]=device.getDestinationIpAddress();
}
Message MensajeValidado = new Message();
for(int i=0;i<20;i++){
    if(TablaRouting[i][1]=device.getDestinationIpAddress()==p2pDevice.deviceAddress){
        MensajeValidado.getData();
    }
}
```

Fig. 7 Codigo tablas ruteo

Se crea las tablas de ruteo en cada dispositivo, cada dispositivo tiene como máximo veinte peers. Se almacenan en dicha tabla el nombre del usuario y dirección IP.

P2pDestinationDevice, una abstracción WifiP2pDevice Enviamos la dirección IP del cliente para el propio cliente desde su GO, para poder almacenarlos en una variable y utilizar esta información para las siguientes necesidades, como lo será la tabla de enrutamiento. Dirección IP GO MainActivity.java

```
@Override
public void onConnectionInfoAvailable(WifiP2pInfo p2pInfo) {
    (...)
    //set Group Owner ip address
    TabFragment.getWifiP2pServicesFragment().setLocalDeviceIpAddress(p2pInfo.groupOwnerAddress.getHostAddress());
    (...)
}
```

Fig. 8 Codigo envio ip cliente

```
Socket clientSocket = socket.accept(); //because now i'm connected with the client/peer device
pool.execute(new ChatManager(clientSocket, handler));
ipAddress = clientSocket.getInetAddress(); //local variable with a get method
```

Fig. 9 Dirección IP del cliente GroupOwnerSocketHandler.java (GO)

```

private void sendAddress(String deviceMacAddress, String name, ChatManager chatManager) {
    if (chatManager != null) {
        InetAddress ipAddress;
        if (socketHandler instanceof GroupOwnerSocketHandler) {
            ipAddress = ((GroupOwnerSocketHandler) socketHandler).getIpAddress();

            Log.d(TAG, "sending message with MAGICADDRESSKEYWORD, with ipaddress= " + ipAddress.getHostAddress());

            chatManager.write((Configuration.PLUSSYMBOLS + Configuration.MAGICADDRESSKEYWORD +
                "___" + deviceMacAddress + "___" + name + "___" + ipAddress.getHostAddress().getBytes());
        } else {
            Log.d(TAG, "sending message with MAGICADDRESSKEYWORD, without ipaddress");
            chatManager.write((Configuration.PLUSSYMBOLS + Configuration.MAGICADDRESSKEYWORD +
                "___" + deviceMacAddress + "___" + name).getBytes());
        }
    }
}
}

```

Fig. 10 MainActivity.java (GO)

```

if (readMessage.contains(Configuration.MAGICADDRESSKEYWORD)) {
    WifiP2pDevice p2pDevice = new WifiP2pDevice();
    p2pDevice.deviceAddress = readMessage.split("___")[1];
    p2pDevice.deviceName = readMessage.split("___")[2];
    P2pDestinationDevice device = new P2pDestinationDevice(p2pDevice);

    if (readMessage.split("___").length == 3) {
        Log.d(TAG, "handleMessage, p2pDevice created with: " + p2pDevice.deviceName + ", " + p2pDevice.deviceAddress);
        manageAddressMessageReception(device);
    } else if (readMessage.split("___").length == 4) {
        device.setDestinationIpAddress(readMessage.split("___")[3]);

        //set client ip address
        TabFragment.getWifiP2pServicesFragment().setLocalDeviceIpAddress(device.getDestinationIpAddress());

        Log.d(TAG, "handleMessage, p2pDevice created with: " + p2pDevice.deviceName + ", "
            + p2pDevice.deviceAddress + ", " + device.getDestinationIpAddress());
        manageAddressMessageReception(device);
    }
}
}

```

Fig. 11 MainActivity.java (CLIENT)

URL de Github: <https://github.com/Krlozz/WiFiDirect>

## 4 Resultados

Codigo Fuente: <https://github.com/Krlozz/WiFiDirect>

Instalación APK

Se deberá ingresar al gestor de archivos del dispositivo y buscar en la carpeta en la que se encuentre el APK se lo seleccionara y se procederá a la instalación

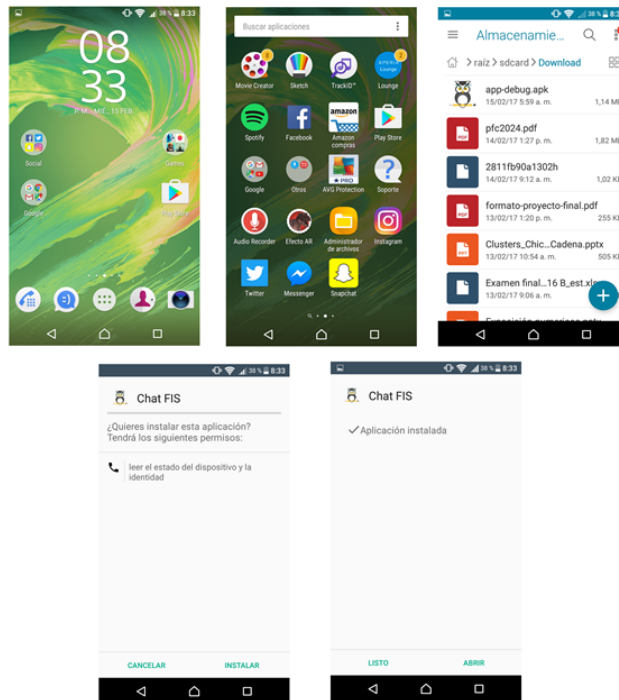


Fig. 12 Instalacion app en dispositivos

Una vez que la aplicación se haya instalado lo que se hará es acceder a ella en cada uno de los dispositivos



Fig. 13 Dispositivos

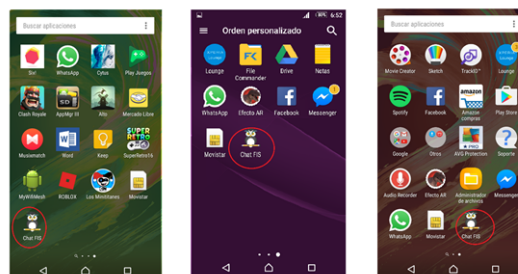


Fig. 14 Verificacion instalacion app



Una vez iniciada la aplicación nos aparecerá el nombre de nuestro dispositivo así como la dirección MAC, también nos mostrara que la IP todavía no ha sido asignada, esto debido a que la IP se asigna cuando se establezca conexión con un dispositivo.

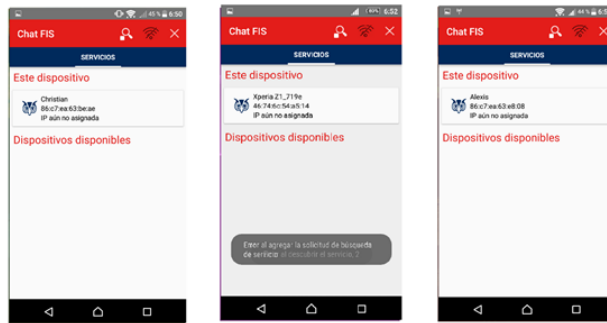


Fig. 15 App corriendo en los dispositivos

La aplicación buscara los dispositivos cercanos que se encuentren disponibles para establecer la conexión.

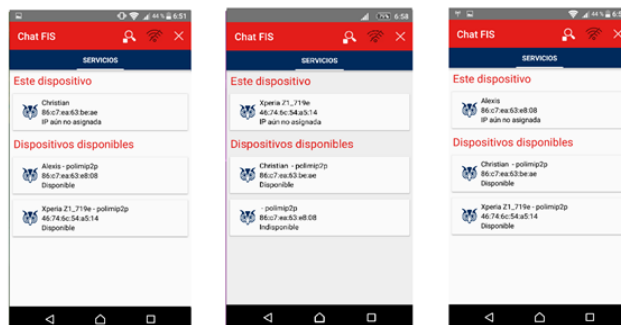


Fig. 16 Búsqueda dispositivos

Para establecer conexión basta solamente con seleccionar sobre el dispositivo al cual deseamos conectarnos y el chat iniciara, en este caso el primer dispositivo enviara un mensaje al tercer dispositivo pasando por el segundo por lo que los dispositivos contarán con las siguientes distribuciones de IP. Primero: IP1, IP2 Segundo: IP1, IP2, IP3 Tercero: IP2, IP3

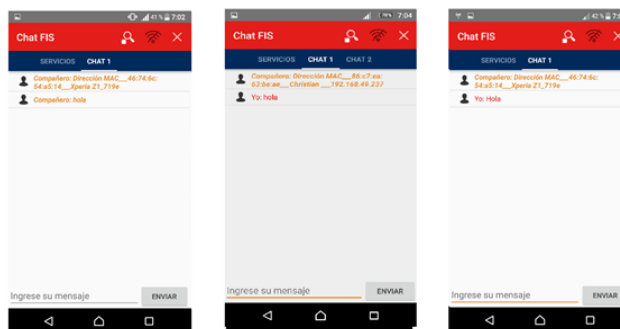


Fig. 17 Conexion y chat

## 5 Conclusiones

Para la creación de la aplicación “Chat FIS” se realizó un estudio con respecto a las conexiones móviles mediante redes inalámbricas como redes P2P y Ad-hoc, y más extensamente sobre la tecnología Wi-Fi Direct en la cual se basa nuestra aplicación.

Se determinó que el uso de Wifi-Direct es una herramienta la cual nos permitirá tener una conexión más rápida que bluetooth a la hora de compartir archivos, originalmente nuestra aplicación fue establecida para establecer una conexión one-to-one con hasta 20 dispositivos pero para lograr una conexión en la que un segundo teléfono actúe como acces point se la tuvo que limitar a tres dispositivos por esta razón se puede decir que la aplicación “Chat Fis” tendría una mejor utilidad al momento de establecer conexión con varios dispositivos debido a que es más viable su uso como chat.

Con la aplacion logramos crear una app de mensajeria que no necesita de internet para funcionar aunque si de conexion wifi, lo cual podria servi de alternativa para comunicarse

Se logro resolver los principales problemas con esta aplicacion que fueron conocer el comportamiento del wifi direct en android y poderlo manejar a nuestro gusto

## 6 Bibliografia

<https://github.com/Krlozz/WiFiDirect>  
[http://oa.upm.es/38862/3/TFG\\_ALMUDENA\\_MONTOYA\\_FERMOSEL.pdf](http://oa.upm.es/38862/3/TFG_ALMUDENA_MONTOYA_FERMOSEL.pdf)  
<https://developer.android.com/training/connect-devices-wirelessly/wifi-direct.html#receiver>  
<https://developer.android.com/guide/topics/connectivity/wifip2p.html>  
<https://openwebinars.net/blog/generar-apk-android-studio/>  
<https://www.adslzone.net/redes/wireless/envia-archivos-por-wi-fi-direct-entre-dispositivos-android-con-wifi-shoot/>  
<https://developer.android.com/training/connect-devices-wirelessly/wifi-direct.html>  
<https://developer.android.com/training/connect-devices-wirelessly/nsd-wifi-direct.html>  
<https://developer.android.com/training/connect-devices-wirelessly/wifi-direct.html>  
<https://developer.android.com/studio/install.html?hl=es-419>  
<http://www.wi-fi.org/discover-wi-fi/wi-fi-direct>